IEEE Open Journal of Vehicular Technology

Received 18 September 2026; accepted 7 October 2026. Date of publication 9 October 2026;
date of current version 27 October 2026. The review of this article was coordinated by Editor Zhonghao Lyu.

Digital Object Identifier 10.1109/OJVT.2026.3742312

An AI-Driven IoT Solution for Public Transport Quality Monitoring and Driving Style Estimation

SIMONE PORCU [1,2] (Member, IEEE), ALESSANDRO FLORIS [1,2] (Senior Member, IEEE), AND LUIGI ATZORI [1,2] (Senior Member, IEEE)

[1]Department of Electrical and Electronic Engineering (DIEE), University of Cagliari, 09123 Cagliari, Italy
[2]National Inter-University Consortium for Telecommunications (CNIT), University of Cagliari, 09123 Cagliari, Italy

CORRESPONDING AUTHOR: SIMONE PORCU (e-mail: simone.porcu@unica.it).

This work was supported by the European Union through the Italian National Recovery and Resilience Plan of NextGenerationEU, "Sustainable Mobility Center" Centro Nazionale per la Mobilità Sostenibile, CNMS, under Grant CN_00000023.
This article has supplementary downloadable material available at https://doi.org/10.1109/OJVT.2026.3742312, provided by the authors.

ABSTRACT Elevating passenger satisfaction in public transit is crucial for incentivizing sustainable mobility and mitigating urban greenhouse gas emissions. To address this challenge, this study introduces an innovative Transport Quality Monitoring System (TQMS) framework that synergizes Internet of Things (IoT) and Artificial Intelligence (AI) to collect and evaluate both empirical and user-perceived quality metrics systematically. The proposed TQMS features a tri-centric architecture spanning hardware, edge, and cloud levels, embedding human-centric feedback directly into the digital replication of the service. Operationally, vehicular telemetry captured via dedicated on-board sensors provides objective insights into driving dynamics, while qualitative passenger feedback is gathered via smartphones. This heterogeneous data is subsequently synthesized at the Edge Layer by the proposed IoT solution running intelligent analytics for real-time quality assessment and anomaly detection. The system's performance was evaluated through empirical field tests conducted on operational urban bus lines, involving 60 participants, 1,200 ride segments, and 20 lines. For predictive modeling, we present a one-vs-rest neural ensemble optimized with a novel temporal coherence loss function designed to minimize jitter in consecutive estimation intervals. Empirical results, obtained under subject-independent, route-independent, and combined subject-and-route evaluation protocols with fold-wise feature selection to prevent data leakage, demonstrate that this tailored loss function on the collected dataset consistently improves upon standard class-weighted Binary Cross Entropy (BCE), increasing macro-F1 by 0.06–0.07 and enhancing the weighted-average F1-score for overall transport service and driving style by up to 8.2% across five distinct quality dimensions, with the largest gains observed on the least represented quality classes.

INDEX TERMS Public transport service quality, Internet of Things, artificial intelligence, user perceived quality, smart mobility, sustainability.

I. INTRODUCTION

Global warming has become one of the most pressing challenges of our time, with 2023 recorded as the hottest year on record, exceeding pre-industrial temperatures by 1.48°C [1]. The rapid rise in global temperatures has led to intensified climate impacts, including severe weather events, ecological disruptions, and economic challenges. A primary driver of this warming is increased greenhouse gas emissions, particularly carbon dioxide from fossil fuel combustion. Major consumers of fossil fuels include energy production companies, which rely heavily on coal, oil, and natural gas to generate electricity and heat. Industrial processes are also significant contributors, as sectors like cement production, steel manufacturing, and chemical processing consume large quantities of fossil fuels

© 2026 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

as energy sources. Additionally, residential and commercial heating and cooling in many regions further add to global emissions.

The transportation sector, another major emitter, accounts for approximately 23% of global energy-related CO_2 emissions [2]. Among these, private vehicles play a predominant role due to their inefficiency and widespread use. In contrast, public transportation systems, such as buses and trains, embody the principles of sharing and collectivity, offering a sustainable alternative to the individualism of private car travel. By consolidating multiple passengers into a single vehicle, public transportation inherently reduces the number of cars on the road, alleviating traffic congestion and decreasing overall fuel consumption and emissions per capita. Not only does this collective approach to mobility lower the carbon footprint of urban travel, but it also highlights the potential for public transit to address broader societal challenges, such as resource efficiency and equitable access to transportation.

To realize these environmental benefits, widespread adoption of public transportation is essential. However, commuters' willingness to switch from private vehicles to public transit depends heavily on service quality. Indeed, factors such as punctuality, comfort, cleanliness, and driving behavior significantly influence public transportation usage and play a crucial role in making it attractive to citizens [3], [4]. Thus, the attention of most public transport companies has shifted from the supply side (i.e., service provision) to the demand side, whose main objective is to improve service quality [5]. Nevertheless, the enhancement of service quality necessitates its continuous measurement and monitoring, enabling the prompt identification and remediation of factors contributing to passenger displeasure. To this aim, literature studies emphasyze the importance of collecting both subjective (passengers' feedback) and objective (routes, timetables, vehicle telemetry) data to train accurate models that can estimate transit service quality [6], [7], [8], [9]. However, these solutions are often restricted to driving-style or ride-comfort estimation. Still, a complete solution for collecting and processing objective and subjective indicators to estimate service quality during the public transport operational phase is missing in the literature.

To address this issue, this paper proposes a Transport Quality Monitoring System (TQMS) architecture that leverages Internet of Things (IoT) and Artificial Intelligence (AI) technologies to collect and process objective and subjective transport data for the estimation of driving style and transport service quality. We believe that the proposed TQMS solution and functionalities could be easily integrated into Digital Twin (DT) solutions, which are increasingly deployed in the transport services domain as a powerful tool for decision support. DTs enable a continuously updated digital version of the physical world and tools for what-if analysis and decision-support simulation. By adding the proposed TQMS, the resulting DT solutions would be able to incorporate quality considerations across all related simulations and analyses. While the adoption of DTs and AI in the public transport scenario is quite widespread, their utilization is typically focused on operational domains, such as fleet management, traffic and route optimization, and diagnostics [10], [11], [12], [13], [14], [15], [16]. All these solutions are often proposed for adoption in decision-support tools for transport authorities and operators, but they underestimate the importance of digitalizing user-perceived quality, a dimension that is absent from most digital representations of transport services.

By giving the possibility to integrate the user-perceived quality into the DT framework, TQMS provides advanced monitoring capabilities across hardware, edge, and cloud layers, which enables transport operators to continuously assess service quality and pinpoint the root causes of any deficiencies, facilitating timely and effective corrective actions. Moreover, TQMS integrates transport service and driving style quality estimation models, implementing a one-vs-rest neural ensemble with a novel temporal coherence loss, which reduces prediction jitter between consecutive estimation windows. The proposed loss adds a gated penalty on score changes that compensates for small fluctuations of sensor measurements due to sudden variations of bus dynamics by enhancing the quality estimation performance.

The major contributions of the paper are the following:

- We introduce a novel IoT three-layer architecture, which can be exploited to integrate the user-perceived quality into the DT framework and leverages AI-based algorithms to provide transport companies with driving style and service quality monitoring and alerting services. These services are powered by both objective and subjective data collected from transport operations.
- We propose a hardware box to measure orientation-related data of the bus to estimate the driving style, and we illustrate the design and development of a smartphone app to collect subjective feedback from the user concerning driving style and service quality.
- We show the experimental tests that we have conducted on a real-world public transport scenario in the city of Cagliari to collect objective (orientation sensor) and subjective (user feedback) data concerning transport data collected from 60 participants for 1,200 ride segments on 20 different bus lines.
- We compare the estimation performance of the one-vs-rest neural ensemble trained using the proposed temporal loss and the class-weighted Binary Cross Entropy (BCE) loss. To achieve robust results, we conducted an ablation study using three group-aware evaluation protocols: Leave-One-Line-Out (LOLO) to assess route transfer, Leave-One-Person-Out (LOPO) to assess subject transfer, and a combined subject-and-route protocol.

The remainder of this paper is organized as follows. Section II reviews related work. Section III provides an overview of TQMS, whereas its layer functionalities are presented in Section IV. Experimental tests and results are discussed in Section V, while Section VI analyzes challenges and limitations of TQMS. Finally, Section VII concludes.

II. RELATED WORK

This section reviews related work on assessing transport service quality and the utilization of DT and AI technologies in transportation systems.

A. TRANSPORT SERVICE QUALITY ASSESSMENT

In recent times, many public transport companies have shifted their focus from simply providing services to better understanding and meeting passenger needs. As service quality becomes increasingly important to everyone involved in public transportation, there is renewed interest in exploring the passenger's point of view. This perspective is being studied in greater depth to encourage the wider use of public transit and, over time, to reduce reliance on private vehicles [5]. Ad hoc studies involved academic and practitioner experts on public transport, who were asked to rate the importance of dozens of indicators describing the quality of public transit services. Then, statistical analysis methods were applied to process the opinion data and identify the set of indicators that contribute to passengers' overall satisfaction and that are crucial for monitoring the quality of public transport services [3], [17], [18]. Among the most important quality indicators were found availability, punctuality, and on-board bus ride comfort, which includes cleanliness, seating and personal space. These indicators provide informative signals of excellence and criticalities in transit services, represent a strategic aspect of desired quality, and are a fundamental input in the monitoring of transit service quality [4].

The challenging part for public transport operators is the measurement of these service quality indicators, which can be commonly conducted using a subjective, objective, and combined approach [19]. Subjective approaches rely on passengers' feedback collected through questionnaires (on-board or online) or face-to-face interviews (on-board). In [5], questionnaires were delivered to a sample of passengers traveling on a particular route to assess overall satisfaction and the importance of the surveyed service quality dimensions for an Italian public bus operator. The questionnaire was phrased in commonly understood language for average passengers and designed to be completed during a 10-minute bus journey due to the limited interest and concentration of respondents. A similar study was conducted in [20], with a particular focus on passengers over 65 years old to investigate whether the local public transport service (Italian city) satisfies their expectations. An online survey was used in [18] to collect data from a total of 481 respondents (customers of the bus transport services of the four major Pakistani cities), who were asked to rate the importance and performance of selected service quality dimensions for the used public transport service. A questionnaire survey was used in [21] to collect all the important and relevant qualitative characteristics of the city bus service in Guwahati, the largest city of Northeast India. While the subjective approach is highly reliable for revealing the main factors influencing customer satisfaction with bus services, it is also time- and cost-consuming, and it only reflects the specific moment (period of time) during which customers are asked about service quality. Moreover, subjective ratings are well known to be easily affected by bias due to the question, rating scale, or personal aspects.

The objective approach leverages objective measures that can undoubtedly provide an unbiased evaluation of bus service quality. These measurements may regard bus punctuality, service regularity, and on-board bus ride comfort [5], [19]. The study in [22] examined the impact of vibrations on the allowable vibration exposure time for intercity bus users. The analysis revealed that passengers on the front and rear overhangs have very short allowed exposure times, and that vibration time could be extended by proper selection of seat oscillatory parameters to improve passenger comfort. A three-dimensional human–bus–road coupled vibration system is proposed in [23], including the bus model, the parallel biomechanical human model with 2 degrees-of-freedom, and road surface roughness condition. The proposed system aims to study the dynamic response of the system and the comfort evaluation of the human body, which is related to the interaction between vehicles and the asphalt road. While the objective approach provides an unbiased evaluation of specific aspects of transit service quality, there is a lack of consideration of these measurements from the user's perspective, which is important since public transport companies must be aware of passengers' points of view.

The combined consideration of subjective and objective measurements is the highly recommended approach because it allows companies to effectively quantify the relationship between objective metrics and subjective quality perception. An artificial neural network (ANN) is proposed in [6] to estimate real-time ride comfort of bus passengers based on measured vehicle-related parameters (e.g., speed, acceleration), passenger-related features (e.g., posture, location), and ride comfort index (vibration values). Passenger rating (23 people involved) was collected from a mobile application as the ground-truth data for ride comfort. The ANN model achieved an R-value of 0.83 for ride comfort prediction. In [7], different objective comfort thresholds were identified to associate passengers' experience comfort conditions with the driving style of the bus drivers. The proposed thresholds were validated on a real case study using approximately 300 subjective judgments and 200,000 pieces of objective kinematic data collected on several routes of an Italian bus operator. In [8], the user's feedback from 18 real driving tests was collected to analyze the correlation between factors induced by motion sickness and the driving vehicle behavior. Six degrees of freedom motion parameters were obtained by the accelerometer of the smartphone. A prediction model of motion sickness duration was defined based on the driving status of vehicles, showing it can be effectively predicted (79.8%) by the vehicle's lateral acceleration, roll, and pitch angular velocity indicators. The study in [9] collected a large number (755) of on-board interviews to evaluate the vibrational comfort of different passengers and analyze its correlation with bus-lane geometry parameters. The results show that travel speeds have

significant correlation values, while there is a moderate inverse correlation between the vibrational level and the age of the passengers. Indeed, the younger age groups tend to have more severe judgments, attributable to their higher demand for comfort.

These studies highlight the importance of jointly collecting both subjective and objective data to train accurate models capable of estimating driving style and transit service quality on public buses. Some solutions have already been proposed in the literature specifically targeting the estimation of on-board comfort based on bus driving style, typically leveraging inertial sensor data to characterize harsh braking, acceleration, and cornering events and correlating them with comfort indices. However, these contributions tend to focus narrowly on a single dimension of the problem, i.e., driving-style-induced discomfort, without addressing the broader and more complex set of factors that determine overall service quality as perceived by passengers, such as crowding, punctuality, cleanliness, or route-specific conditions.

To date, a complete architectural solution capable of systematically collecting diverse types of objective measurements alongside corresponding subjective scores, and integrating them to estimate service quality throughout the public transport operational phase, is still missing in the literature. Existing approaches remain fragmented, addressing isolated aspects of the problem (e.g., driving comfort or vehicle occupancy) in isolation, rather than providing a unified framework that spans data acquisition, synchronization of heterogeneous data streams, and quality modeling. This gap motivates the need for an end-to-end solution capable of bridging objective sensing infrastructure and subjective quality assessment within a single operational architecture.

B. DTS AND AI IN TRANSPORTATION SYSTEMS

Recent research has increasingly focused on the integration of DT technology into transportation systems to enhance intelligence, monitoring, and lifecycle management. DTs in transportation are expected to optimize performance, enhance passenger experience, and support sustainable urban mobility through improved operational accuracy and energy efficiency. Beyond architectures [24], [25], which determine how heterogeneous data, models, and services are orchestrated, the value of DTs in transportation lies in their ability to embed AI to support prediction and decision-making [11]. Particular applications of DTs have been focused on specific micro-areas of the transportation infrastructure, such as railways, highways, bridges, and tunnels [26], [27]. Other DT applications include autonomous driving, electric vehicles, and intelligent traffic management systems [28], [29], [30].

Concerning public transportation systems, the main role of DTs is to offer virtual representations of physical systems by integrating real-time data from various sources, which can enable predictive analytics, optimize operations, and improve the overall performance of public transportation networks. The current applications of DTs in public bus-based transportation include bus fleet management, traffic and route optimization, passenger experience enhancement, and physical modeling of buses [10], [11]. A DT implementation to simulate and optimize the bus system in the city of Lisbon is proposed in [31]. By feeding the DT with data containing ticketing and General Transit Feed Specification (GTFS) information, simulation results have shown that schedule adjustments and demand-based fleet allocations yield substantial improvements in both service quality and operational efficiency. An implementation of a DT of the public bus system of the city of Malaga is presented in [12]. High-level UML models were used to design bus, journey, and passenger entities, while open data were used to get information concerning bus utilisation, including data about routes, stops, timetables, and bus position. A DT of a public transportation system in Badalona is proposed in [13] for obtaining an in-depth understanding of the bus dynamics. A genetic algorithm was used to find the most suitable configurations for simulating the traffic in a city based on real data. Results show that the proposed DT can accurately reproduce the real traffic and the bus schedule. The study in [14] implemented a DT to simulate transit demand and bus operation, and to calibrate bus speed to respect the bus schedule. Regarding AI integration in DT systems for public transportation, the majority of literature studies integrate real-world operational data sourced from public transportation systems (such as passenger flow, bus position and speed) to provide data-driven predictive, diagnostic, and optimization functions, such as demand forecasting, fleet management, traffic prediction, incident analysis, and route optimization [15], [16]. While these studies leveraged DTs and AI to model dynamic operations in transit public services, they tend to remain tightly scoped to narrow operational or engineering domains, with no consideration of transit service quality and the passenger's perspective.

C. RESEARCH MOTIVATIONS

The current applications of DTs in public bus-based transportation include feeding the DT with diverse data (e.g., ticketing, transit feed, routes, timetables, etc.) to implement AI-based simulations aimed at fleet management, traffic and route optimization, and bus diagnostics. All these solutions help operators reduce congestion, optimize fleet operations, and better match service supply with passenger demand. However, based on the related work discussed in this section, and to the best of our knowledge, no studies have considered leveraging the potentialities of DTs and AI to measure transit service quality of bus-based public transport. Most of researches on transport service quality provides ad hoc solutions to estimate on-board comfort on buses based on driving style, but a general solution for transport service quality has not been proposed yet. The solutions proposed in the literature underestimate the importance of digitalizing user-perceived quality, a dimension that is absent from most digital representations of transport services.

Moreover, the recent review on DTs in public bus transport identified AI integration, evaluation metrics, and user-centered design processes as three concrete recommendations

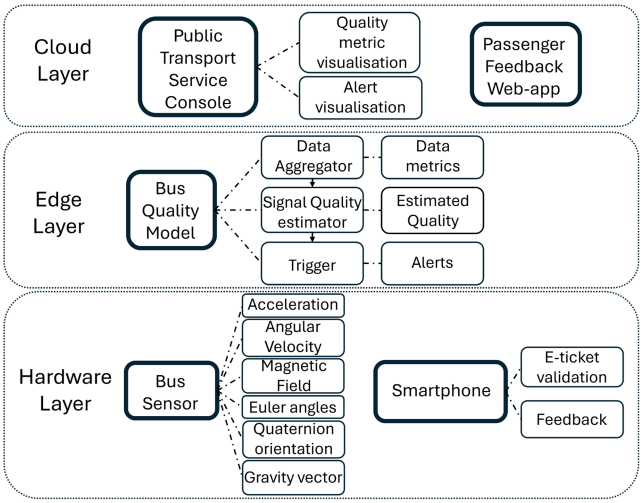

FIGURE 1. The architecture of the proposed TQMS.

(among a total of 8) for advancing DT research in bus-based public transport systems [11]. These reasons motivated the development of the proposed TQMS, whose architecture is centered on the Bus Quality Model acting as the virtual counterpart of the physical bus with reference to the user-perceived quality. The Bus Quality Model can estimate the transport service quality of the bus by considering both the objective measurements from the sensors equipped on the bus and the subjective perspective of the passengers. To include such a user-centered design, we implemented a Passenger Feedback Web-App that collects the feedback of the passengers. Finally, we propose a novel AI-based estimation model for transport service quality that leverages a temporal coherence loss to tailor the temporality of the collected objective and subjective measurements.

III. OVERVIEW OF TQMS

The proposed TQMS is designed to support public transport service providers in the monitoring and management of transport service quality. To reach this objective, TQMS relies on a three-layer architecture (hardware, edge, and cloud layers, as shown in Fig. 1) centered on the Bus Quality Model, which acts as the virtual counterpart of the physical bus with a specific focus on the quality aspects. It is implemented at the Edge Layer infrastructure, where it can quickly communicate and exchange data with the Hardware Layer (where objective measures from the physical bus are collected) and the Cloud Layer (where feedback from bus passengers is collected) to feed input data for service quality estimators leveraging AI-based algorithms. Moreover, the Bus Quality Model has the potential to receive (e.g., through APIs) and aggregate data from external services providing crucial information for enhancing service quality estimation. For instance, data concerning bus punctuality, which is typically monitored by transport service providers, or data from specific systems developed by third-party companies measuring the bus passenger occupancy. Thus, the Bus Quality Model acts as a collector and aggregator for all kinds of information (and historical records) concerning the physical bus it is associated with, which the developed DT can leverage to accurately estimate the service quality perceived by the bus's passengers. Such a quality monitoring service enables the service providers to receive alerts when quality deficiencies are detected, which can drive specific actions depending on the cause of the quality decrease.

A. SYSTEM ARCHITECTURE

Fig. 1 illustrates the three-layer architecture for TQMS. The Hardware Layer forms the foundation of the TQMS architecture by including sensors and data transmission units installed in the buses to facilitate continuous data acquisition processes. Orientation sensors, such as accelerometers, gyroscopes, and magnetometers, capture driving dynamics to estimate passenger comfort. Moreover, the passenger's smartphone is included in this layer because it is the hardware device that enables TQMS to collect the data concerning e-ticket validation activities and subjective ratings on the passenger's perceived driving style and transport quality.

The Edge Layer includes and leverages the functionalities of the Bus Quality Model for data processing and analysis of the acquired bus operational state information received from the Hardware Layer. The analyzed data is used to train deep learning neural networks aimed at estimating bus driving style and transport service quality, and to generate alerts in cases where these do not reach adequate levels. The bus driving style indicates the passenger's ride comfort due to the driving behavior, whereas the transport service quality concerns the overall quality perceived by the passenger during the journey, including ride comfort and further aspects, such as punctuality, cleanliness, etc.

Each bus is represented by a unique model, which integrates three main functional modules:

- *Data Aggregator:* This module preprocesses the raw sensor data received from the Hardware Layer to compute statistical and temporal metrics, which are correlated with the subjective passenger feedback to capture patterns related to driving style and transport service quality. Data metrics are generated and sent to the Signal Quality Estimator, and made available to external systems when needed.
- *Signal Quality Estimator:* This is the intelligent core of the model, which trains driving style and transport service quality estimators from the computed statistical and temporal metrics. The developed estimators combine an ensemble of neural networks with a one-vs-rest strategy to enhance multi-class classification performance. The estimated quality is made available to external systems and also sent to the Trigger module.
- *Trigger:* This module monitors the estimated driving style and transport service quality received from the Signal Quality Estimator module and verifies whether these scores fall below predefined thresholds. When a

threshold is exceeded, it generates alerts to the Cloud Layer, including the timestamp, the affected metric, and the associated score.

Finally, the Cloud Layer hosts two centralized modules: the Public Transport Service (PTS) Console and the Passenger Feedback Web-App. The former is intended to provide a dashboard to receive and visualize the alerts generated by the Trigger module, whereas the latter allows passengers to provide subjective feedback about their journey quality through their smartphone device.

B. REFERENCE SCENARIO

To illustrate the workflow of TQMS, we describe a typical scenario involving a passenger travelling on a bus equipped with TQMS. When the passenger boards the bus and validates the e-ticket using his smartphone (where the Passenger Feedback Web-App is installed), TQMS records the bus ID, e-ticket ID, and validation timestamp. Simultaneously, the sensor board embedded in the vehicle measures objective data related to the driving style. These data streams are stored in the Bus Quality Model, where the Data Aggregator module preprocesses them through data cleaning, data normalization, and data structuring processes. The preprocessed data is then used by the Signal Quality Estimator module to predict the driving style and the transport service quality. If either metric falls below predefined acceptability thresholds, the Trigger module is activated and generates an alert towards the PTS Console. Upon receiving an alert, the PTS Console administrator can investigate the underlying causes of the quality decline, whether related to driving style or other factors, and implement appropriate corrective actions.

When the passenger disembarks at his destination, he notifies TQMS by marking the end of the journey through the smartphone's app. The difference between disembarkation and ticket validation defines the precise duration of the passenger's journey. Moreover, through the Passenger Feedback Web-App, the passenger can rate the perceived driving style and transport service quality. This feedback will be associated with the passenger's ticket ID, bus ID and the corresponding objective data measured on the bus during the journey. Once the feedback is received, the Signal Quality Estimator module requests the relevant data window from the Data Aggregator module. The resulting data, combining objective measurements and subjective feedback, is used to update the driving style and service quality prediction models. This continuous learning process allows the model to refine its accuracy and remain aligned with modifications of passenger perceptions. By closing the loop between passenger feedback and on-board analytics, the TQMS solution enables proactive service monitoring and targeted interventions to enhance both operational efficiency and passenger satisfaction.

C. SYSTEM ONTOLOGY

The TQMS ontology, illustrated in Fig. 2, defines a formal conceptual model that captures the structure of the system, the roles of its components and their interactions, highlighting

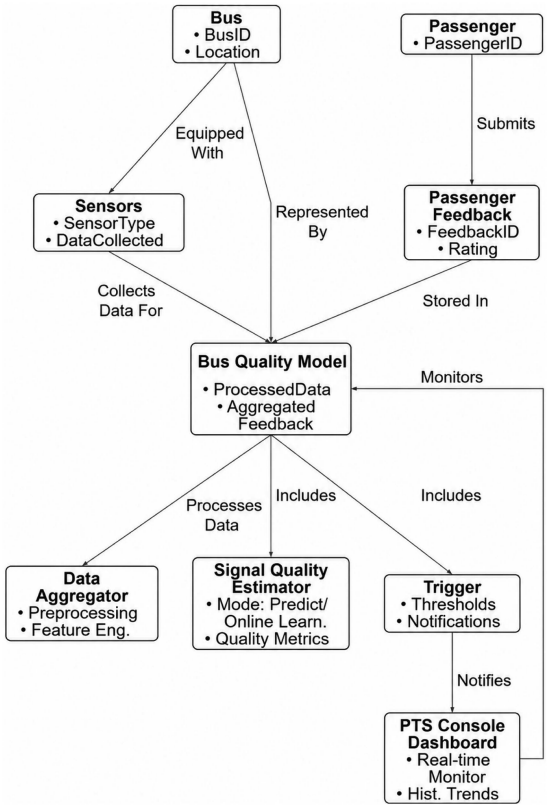

FIGURE 2. The TQMS ontology, illustrating relationships between key entities.

the relationships between physical entities (e.g., buses, sensors, passengers) and digital abstractions (e.g., quality model, feedback, alarms), enabling semantic consistency across hardware, edge, and cloud layers.

Each bus is represented by a Bus Quality Model that collects objective measurements from the bus's on-board sensors and subjective passenger feedback from their smartphone app. The Bus Quality Model includes modules responsible for data processing and analysis, quality estimation, and event detection. Passengers interact with the system through feedback submissions, which are linked to specific journeys via identifiers and timestamps. When quality thresholds are violated, the Bus Quality Model notifies the PTS Console for operator intervention.

Table 1 complements Fig. 2 by summarizing each entity's key attributes and their role in the system. This ontology supports modularity, data traceability, and automated reasoning within TQMS, allowing consistent integration of multimodal data and dynamic service adaptation.

D. THE TQMS IN A DT ARCHITECTURE

The proposed TQMS is an IoT solution devoted to the monitoring and prediction of bus transport service quality. It includes sensors, communication components, quality bus modeling, prediction modules and functionalities for quality visualization and modeling. These components and related

TABLE 1. Attributes and Roles of Key Entities in the TQMS Ontology

Entity	Attributes	Roles/Relationships
Bus	• *BusID*: Unique identifier for the bus. • *Location*: GPS coordinates.	• *EquippedWith* **Sensors**. • *RepresentedBy* **Bus Quality Model**.
Sensors	• *SensorType*: Type of sensor (e.g., accelerometer). • *DataCollected*: Data such as acceleration, vibrations.	• *CollectsDataFor* **Bus Quality Model**.
Bus Quality Model	• *ProcessedData*: Preprocessed data from sensors and subjective feedback.	• *ProcessesDataFrom* **Sensors** and **Passenger Feedback**. • *Includes* **Data Aggregator**, **Signal Quality Estimator**, **Trigger**.
Data Aggregator module	• *AggregationLogic*: Method for data preprocessing (e.g., normalization, cleaning).	• *ProcessesDataFor* **Signal Quality Estimator**.
Signal Quality Estimator module	• *ModelType*: Type of deep learning model used. • *Mode*: Online learning or Prediction.	• *TrainsOn* **Passenger Feedback** and **Processed Data**. • *Predicts* **Quality Metrics**.
Trigger module	• *Thresholds*: Acceptability thresholds for quality metrics.	• *Activates* **Quality Alarm Notification**.
Passenger Feedback	• *FeedbackID*: Unique identifier for feedback. • *Rating*: Passenger-provided score for the journey (e.g., driving style, service quality).	• *SubmittedBy* **Passenger**. • *StoredIn* **Bus Quality Model**.
Passenger	• *PassengerID*: Unique identifier for the passenger.	• *Submits* **Passenger Feedback**.
PTS Console Dashboard	• *DashboardType*: Current or Historical.	• *Monitors* **Fleet Data**.

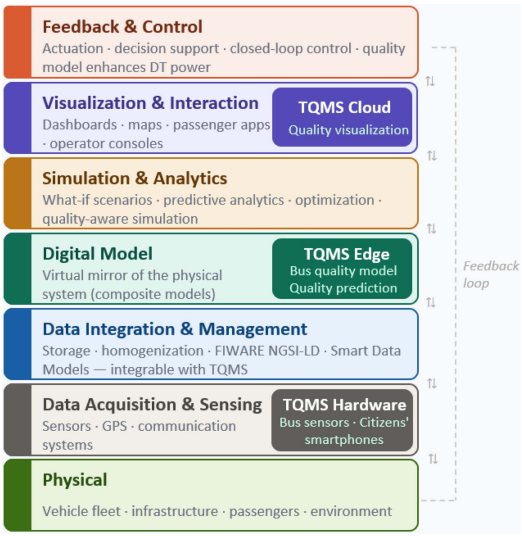

FIGURE 3. Mapping of TQMS components into the DT layer stack.

capabilities can be integrated into DT solutions, which are increasingly deployed in the transport services domain. Indeed, a complete DT solution also includes the following functionalities: continuous synchronization between physical and virtual states; behavior prediction or simulation under alternative scenarios; model updating and bidirectional feedback/control; lifecycle representation of the bus or service; what-if analysis or decision-support simulation. By adding the proposed TQMS, the resulting DT solutions would be able to incorporate quality considerations across all related simulations and analyses.

Fig. 3 shows a reference DT architecture [32] and where the TQMS components are located. The Physical Layer is where the physical world that the DT mirrors is located. It includes all the public transport assets, e.g., the vehicle fleet, the infrastructure they operate within, as well as the passengers. This layer is not passive, but it is the source of all real-world events and the ultimate target of any corrective or optimization action generated by the DT. The Data Acquisition and Sensing Layer relies on a dense network of sensors and communication systems (e.g., GPS receivers, accelerometers, door sensors, and passenger-counting systems). These continuously generate operational data, which are sent via different communication technologies and used by the upper layers, enabling continuous synchronization between physical and virtual states. This is the one where the TQMS Hardware Layer is located, with all the Bus Sensors and Citizens' Smartphones. The Data Integration and Management Layer is where a robust infrastructure for storing, organizing, and harmonizing it is located. This layer also addresses semantic interoperability: standards such as FIWARE NGSI-LD and the associated Smart Data Models for Smart Cities define common vocabularies and data structures, ensuring that data from heterogeneous sources can be understood and combined consistently. The proposed architecture does not address this aspect. However, components of this layer can be easily integrated in the proposed solution (or our proposal can be integrated in a DT with these layer components) and benefit from relevant advantages by exploiting state-of-the-art solutions. Proper interfaces enable data collection from the proposed hardware components to build a homogenized dataset that can also be used for other purposes.

The Digital Model Layer is the heart of the DT and represents the virtual representation that mirrors the state of the physical system at any given moment. It is not a single model but a composite of several complementary representations operating at different levels of abstraction. This is where the proposed Edge Layer components are located. The collected data is used to build the Digital Model (our Bus Quality Model) to predict achieved quality levels. The Simulation and Analytics Layer is where the DT's predictive and prescriptive power is realized. What-if simulation allows operators and planners to test hypothetical scenarios (e.g., a new route configuration, a modified timetable, a special event) without risking real-world operations. Predictive analytics use the models to anticipate future states. In our proposal, this aspect has not been implemented by the proposed Bus Quality

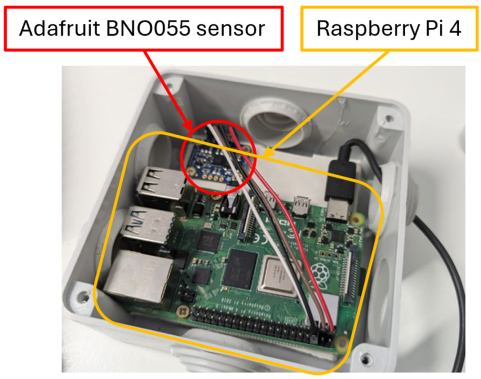

FIGURE 4. On-board device: Raspberry Pi 4 equipped with the Adafruit BNO055 absolute orientation sensor.

model. However, the developed quality models can be used to significantly enhance the simulation analysis capabilities. The Visualization and Interaction Layer is where the insights generated by the DT must be accessible to those who need them, whether they are control room operators, network planners, maintenance engineers, or passengers. This is the layer where the proposed Cloud Layer functionalities are located. Finally, the Feedback and Control Layer is probably the most distinctive component of a DT, where the feedback loop closes the cycle between the digital and physical domains. Indeed, unlike a static simulation model, a DT is designed to act.

IV. THE TQMS LAYER FUNCTIONALITIES

This section provides additional details on the functionalities and implementation of the TQMS solution.

A. HARDWARE LAYER

The Hardware Layer leverages an electronic board, shown in Fig. 4, composed of a Raspberry Pi 4 equipped with the Adafruit BNO055 absolute orientation sensor, which integrates a 3-axis accelerometer, a gyroscope, and a magnetometer. Specifically, the following data is captured:

- **Acceleration** along the three spatial axes (x, y, z) in m/s^2. Both gravity and linear motion acceleration are measured, allowing the system to analyze forces generated by braking, turning, and road surface conditions.
- **Angular velocity** along the three axes of rotation speed in rad/s.
- **Magnetic field strength** along the three axes of magnetic field sensing in μT.
- **Euler angles** (roll, pitch, yaw) representing absolute orientation.
- **Quaternion orientation** for singularity-free 3D rotation.
- **Gravity vector** along the three axes of gravitational acceleration in m/s^2.

Thanks to this set of comprehensive data, the BNO055 sensor is capable of detecting even the most subtle motion and perturbation experienced during bus journeys, such as braking, turning, or encountering uneven road surfaces. Moreover, events such as bumps, sharp turns, and instability can be detected. Thus, this data provides a comprehensive understanding of the vehicle's driving style, forming the foundation for evaluating journey quality and triggering alerts when thresholds are exceeded.

The Raspberry Pi acts as the CPU and data transmission unit to aggregate the data collected from the different sensors and to transfer it towards the Edge Layer. A 5G antenna is used to provide the Raspberry Pi with an Internet connection. The Raspberry Pi and the sensors are housed in a custom-fabricated enclosure to ensure portability, security, and consistency of the hardware setup. The board is powered by the bus battery, enabling continuous data collection throughout each journey. To mitigate the well-known temperature drift and bias instability of the BNO055 sensor, a systematic calibration procedure was performed before each test run. Furthermore, the sensor placement in the bus was optimized to minimize vibrations and thermal interference, ensuring data consistency.

B. EDGE LAYER

To orchestrate local services of the Bus Quality Model, the Edge Layer relies on a lightweight server based on the Flask framework, which exposes RESTful APIs that support interaction with the three modules: Data Aggregator, Signal Quality Estimator, and Trigger. Running as a persistent background service, this server ensures the continuous availability of the Bus Quality Model logic and provides an interface for integration with the broader TQMS architecture.

1) DATA AGGREGATOR

The Data Aggregator module leverages a lightweight NoSQL database, optimized for edge deployment, to store the data. Its schema-less structure and minimal resource footprint enable this database for persistent and offline-capable data storage even under constrained connectivity conditions. Three main types of data are collected: raw time-stamped sensor data received from the electronic board, computed statistical and temporal metrics, and passenger feedback.

The raw time-stamped sensor data is preprocessed using the Python language and its libraries (e.g., NumPy and SciPy). A wide range of statistical and temporal metrics is computed to provide a feature-rich summary that describes the physical dynamics of the bus driving style. Statistical metrics characterize the global statistical properties of the signal within a time window with no consideration of the samples' order. These metrics include mean, standard deviation, maximum, minimum, range, and root mean square (RMS), which summarize the magnitude and dispersion of the signal over a fixed time window of length T (with a default value of 5 minutes), allowing for a balance between temporal resolution and stability. Temporal metrics, on the other hand, are designed to capture the evolution of the signal over time, providing insights into its dynamic behavior to achieve a more comprehensive analysis

TABLE 2. Statistical and Temporal Metrics Computed by the Data Aggregator Module

Metric	Definition	Formula
Mean	Average value of all samples.	$\text{mean} = \frac{1}{n}\sum_{i=1}^{n} x_i$
Standard Deviation (std)	Measures how much the values vary from the mean.	$\text{std} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \text{mean})^2}$
Maximum (max)	The highest value between the samples.	$\max(x_1, \ldots, x_n)$
Minimum (min)	The lowest value between the samples.	$\min(x_1, \ldots, x_n)$
Range	Difference between the maximum and minimum values.	$\text{range} = \max - \min$
Root Mean Square (rms)	Quadratic mean.	$\text{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$
Difference Mean (diff mean)	Average change between consecutive samples.	$\text{diff_mean} = \frac{1}{n-1}\sum_{i=1}^{n-1}(x_{i+1} - x_i)$
Difference Standard Deviation (diff std)	Variation in the change between consecutive samples.	$\text{diff_std} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n-1}((x_{i+1} - x_i) - \text{diff_mean})^2}$
Rate of Change	Net change divided by number of steps (samples - 1).	$\text{rate_of_change} = \frac{x_n - x_1}{n-1}$
Peak Count	Number of local maxima in the signal.	$\text{peak_count} = \sum_{i=2}^{n-1}\delta(x_{i-1} < x_i > x_{i+1})$
Peak Mean	Average value at detected peaks.	$\text{peak_mean} = \frac{1}{k}\sum_{i=1}^{k} x_{p_i}$, where p_i are peak indices
Dominant Frequency	Frequency with the highest magnitude in the FFT spectrum.	$\text{dominant_freq} = \arg\max_{f \neq 0} \lvert\mathcal{F}(x)(f)\rvert$
Magnitude Frequency	Magnitude of the dominant frequency.	$\text{freq_magnitude} = \lvert\mathcal{F}(x)(\text{dominant_freq})\rvert$
Autocorrelation	Similarity between a signal and a delayed version of itself. Detects repeating patterns.	$\text{autocorr}(k) = \frac{\sum_{i=1}^{n-k}(x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$
Trend Slope	Measures whether the signal increases, decreases, or stays stable by fitting a linear trend.	$m = \frac{\sum_{i=1}^{n}(t_i - \bar{t})(x_i - \bar{x})}{\sum_{i=1}^{n}(t_i - \bar{t})^2}$
Volatility	Measures short-term variability by averaging the standard deviation across moving windows.	$\sigma_{\text{vol}} = \frac{1}{N}\sum_{i=1}^{N}\sigma_i, \quad \sigma_i = \sqrt{\frac{1}{k}\sum_{j=1}^{k}(x_j - \bar{x}_i)^2}$

of the physical phenomena observed during the bus journey. These metrics include difference mean, difference standard deviation, rate of change, peak count, peak mean, dominant frequency, frequency magnitude, autocorrelation, trend slope, and volatility. These features are sensitive to the order of the samples and are used to capture patterns, oscillations, and variations in the signal. The computed statistical and temporal metrics are summarized in Table 2, whereas Section V-A presents an analysis to identify the most informative features based on their contribution to the estimation models.

Moreover, the Data Aggregator associates passenger feedback received from the Cloud Layer with the corresponding computed temporal and statistical metrics using time-aligned intervals. When multiple passengers provide feedback for the same journey (they were in the same bus at the same time), the Data Aggregator calculates the Mean Opinion Score (MOS) by averaging their ratings, as follows:

$$\text{MOS} = \textit{round}\left(\frac{1}{M}\sum_{j=1}^{M} r_j\right), \tag{1}$$

where r_j is the feedback rating from passenger j, M is the total number of feedback entries, and *round* is the function used to round the MOS to the nearest integer. This process ensures that subjective evaluations are associated with objective measurements, enabling a holistic analysis of journey quality.

2) SIGNAL QUALITY ESTIMATOR

The Signal Quality Estimator module evaluates both driving style and transport service quality using a deep learning approach that combines an ensemble architecture of neural networks with a one-vs-rest strategy to enhance multi-class classification performance. Each class is handled by a separate binary classifier that distinguishes it from all others. As shown in Fig. 5, each classifier is a feedforward neural network with three layers: an input layer, one hidden layer with 64 neurons, and an output layer. The hidden layer uses the ReLU activation function to introduce non-linearity.

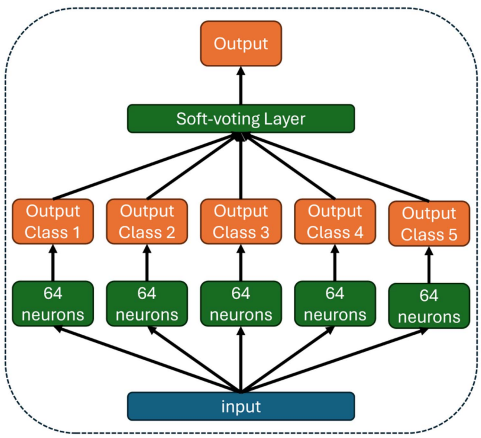

FIGURE 5. Structure of the feedforward neural network used in each binary classifier.

The novelty of this proposed estimation model is that it leverages a temporal coherence loss to discourage large changes between the class scores of two adjacent quality evaluation windows. Indeed, typically, literature estimation models use standard binary cross-entropy loss functions trained on single windows. However, sensors can be noisy and small fluctuations can flip the prediction between consecutive windows even when the ride is calm. This creates jitter in

the middle classes that is not aligned with how people judge quality over the whole ride. Thus, the proposed temporal coherence loss introduces a gate computed from the variance of the input features inside the window to set the strength of the penalty. When the variance is low, the gate is high, and the smoothing is strong. When the variance is high, the gate is low, and the smoothing is weak. This way, the previous score is treated with stopgrad, and the training stays feedforward with mini batches and does not require backpropagation through time.

The network input is a feature vector:

$$\mathbf{x} \in \mathbb{R}^n, \quad \mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T, \tag{2}$$

where n is the number of input features computed by the Data Aggregator on windows of length T.

The hidden layer computes the activation:

$$\mathbf{h} = \phi(\mathbf{z}^{(1)}), \quad \text{where} \quad \mathbf{z}^{(1)} = \mathbf{W}_1\mathbf{x} + \mathbf{b}_1, \tag{3}$$

with $\mathbf{W}_1 \in \mathbb{R}^{64\times n}$ as the weight matrix, $\mathbf{b}_1 \in \mathbb{R}^{64}$ the bias vector, and $\phi(a) = \max(0, a)$ the ReLU activation function applied element-wise.

For the binary problem associated with class j, the output layer produces the scalar logit and the corresponding probability:

$$z_j = \mathbf{W}_{2,j}\mathbf{h} + b_{2,j}, \qquad p_j = \sigma(z_j) = \frac{1}{1+e^{-z_j}}. \tag{4}$$

In total, the ensemble contains k classifiers $\{f_1, f_2, \dots, f_k\}$, one per class. Each classifier outputs a probability $p_j = f_j(\mathbf{x})$, and the final prediction is made via soft voting:

$$\hat{y} = \arg\max_{j\in\{1,\dots,k\}} \alpha_j\, p_j, \tag{5}$$

where α_j is an optional weight assigned to class j (by default $\alpha_j = 1$).

Each binary classifier is trained independently. For the j-th classifier, the binary target is defined as:

$$y_j^{(i)} = \begin{cases} 1 & \text{if } y^{(i)} = j \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

We index training samples with $i = 1, \dots, N$ and time windows with t. There are k classes, and we train one binary classifier for each class $j \in \{1, \dots, k\}$. For sample i, the one-vs-rest label for class j is $y_j^{(i)} \in \{0, 1\}$ and the predicted probability is $p_j^{(i)} \in [0, 1]$. At window t, the network input is $\mathbf{x}_t = [x_{t,1}, \dots, x_{t,n}]^\top \in \mathbb{R}^n$, where n is the number of input features.

To address class imbalance, we use an inverse-frequency weighted binary cross entropy for each class j:

$$\mathcal{L}_j^{\text{BCE}} = -\frac{1}{N}\sum_{i=1}^{N}\Big[w_1^{(j)}\, y_j^{(i)} \log\left(p_j^{(i)}\right) + w_0^{(j)}\left(1 - y_j^{(i)}\right)\log\left(1 - p_j^{(i)}\right)\Big]. \tag{7a}$$

$$w_1^{(j)} = \frac{N}{2\,N_1^{(j)}},\ w_0^{(j)} = \frac{N}{2\,N_0^{(j)}}, \tag{7b}$$

$$N_1^{(j)} = \sum_{i=1}^{N} y_j^{(i)},\ N_0^{(j)} = N - N_1^{(j)}. \tag{7c}$$

Here $N_1^{(j)}$ and $N_0^{(j)}$ are the counts of positive and negative samples for class j.

Predictions are produced at each window t of length T. Let $p_j^{(t)} \in [0, 1]$ be the one-vs-rest score of class j at window t. We define a simple volatility for window t as the variance across the n features:

$$\text{var}_{\text{sens}}(t) \;=\; \frac{1}{n}\sum_{u=1}^{n}\big(x_{t,u} - \bar{x}_t\big)^2, \qquad \bar{x}_t \;=\; \frac{1}{n}\sum_{u=1}^{n} x_{t,u}. \tag{8}$$

We built a gate for each window. It uses two hyperparameters. The first is $m \in [0, 1)$, the minimum gate value that prevents the temporal penalty from vanishing. The second is $\beta \geq 0$, which controls how fast the gate decreases when volatility grows.

$$g_t \;=\; \max\big(m,\ \exp(-\beta\,\text{var}_{\text{sens}}(t))\big)\,. \tag{9}$$

The temporal penalty for class j compares the current score with the previous window:

$$\mathcal{L}_{\text{temp}}^{(j)} \;=\; \mu\sum_{t\in\mathcal{T}} g_t\,\left(p_j^{(t)} - \text{stopgrad}\left(p_j^{(t-1)}\right)\right)^2, \qquad \mu > 0. \tag{10}$$

The index set $\mathcal{T}$ contains windows that have a valid previous window from the same route and the same subject and that appear in the same minibatch. The operator stopgrad$(\cdot)$ blocks gradients through the previous score.

The per-class objective is

$$\mathcal{L}^{(j)} \;=\; \mathcal{L}_{\text{BCE}}^{(j)} \;+\; \mathcal{L}_{\text{temp}}^{(j)}. \tag{11}$$

The ensemble objective sums the per-class objectives

$$\mathcal{L}_{\text{total}} \;=\; \sum_{j=1}^{k}\alpha_j\,\mathcal{L}^{(j)}, \quad \alpha_j \in \mathbb{R} \geq 0. \tag{12}$$

All binary classifiers were trained with the Adam optimizer, using a learning rate of 10^{-3}, a mini-batch size of 32, and an L_2 weight decay of 10^{-4}. Training ran for at most 200 epochs with early stopping on the validation loss, with a patience of 20 epochs and a minimum improvement threshold of 10^{-4}. The weights of the best epoch were restored at the end of training. Each fold was trained with 5 different random seeds, and the reported results were obtained by pooling the corresponding out-of-fold predictions. The temporal term used $\mu = 0.10$, $\beta = 1.0$, and $m = 0.20$. We drew mini-batches with a pair-aware sampler to ensure both elements of a temporally adjacent pair belonged to the same mini-batch and contributed to the temporal penalty.

In all experiments, we used uniform weights $\alpha_j = 1$ for all j. When features are calm, $\text{var}_{\text{sens}}(t)$ is small and g_t is close to one, so the model discourages small jumps. When features are

turbulent, $\text{var}_{\text{sens}}(t)$ is larger and g_t decreases but never below m. We choose μ so that the temporal term is smaller than the cross entropy term on labeled changes, which preserves responsiveness.

The Signal Quality Estimator module runs in two modes: i) Prediction mode: Every interval of length T, the Data Aggregator computes features that are used by the Signal Quality Estimator module to predict the driving style and transport service quality. If these predictions drop below a selected threshold, the Trigger module raises an alert. ii) Online learning mode: When a passenger submits feedback, the system retrieves the corresponding objective trip data from the bus. This feedback is then used to update the model training and improve prediction performance.

3) TRIGGER

The Trigger module operates by continuously monitoring two key outputs from the Signal Quality Estimator module: the estimated driving style and transport service quality scores, which are updated at regular intervals during bus operation. To identify potential issues, this module evaluates whether these scores fall below a predefined threshold, concerning a sufficient perceived service quality (e.g., a score of 3 if a 5-level quality scale is considered).

If either the driving style or transport service quality score breaches this threshold, the Trigger module flags the event as a potential quality issue and generates an alert containing detailed information about the event, including: i) Timestamp: the exact time of detection, ensuring accurate correlation with operational data. ii) Bus ID: the unique identifier of the bus where the quality issue occurred. iii) Affected Metric: the specific quality metric (driving style or transport service quality) that triggered the alert. iv) Score Value: the score value below the threshold, providing quantitative context for the alert.

C. CLOUD LAYER

At the Cloud Layer, the PTS Console is a Web-based dashboard developed with Flask and interactive Plotly visualizations, designed to support operators in monitoring and managing bus fleet-wide performance. Upon login, operators are presented with an overview of active bus lines. Selecting a line reveals the list of operating buses and their corresponding Bus IDs. Each bus profile links to a metrics page displaying the driving style and the transport service quality estimated by the Signal Quality Estimator module. If the Trigger module detects low levels of driving style or transport service quality, alerts are shown on the corresponding metrics page. This allows operators to promptly respond with corrective actions, such as notifying drivers or rescheduling services. The modular architecture ensures that the console remains responsive and scalable as fleet size increases.

The Cloud Layer also includes the Passenger Feedback Web-App, developed with Flask and Bootstrap. This app offers a simple form to assess journey quality and driving style on a 5-level scale with optional free-text comments. Each feedback submission is associated with the corresponding journey using the bus ID and timestamp. This enables precise synchronization between subjective feedback and sensor data collected on the bus. Once received, the feedback is stored and used by the Signal Quality Estimator module (in the Online learning mode) to update the prediction models in order to be continually adapted to the evolving passenger perceived quality. This feedback loop ensures TQMS remains centered on user experience, blending sensor analytics with human perception to support service improvement.

D. DATA SECURITY

TQMS links a validated journey to objective trip data and user feedback. At boarding, it records the bus ID, e-ticket ID, and validation timestamp; at the end of the trip, the Passenger Feedback Web-App associates the rating to the journey using identifiers and time alignment, and the Signal Quality Estimator uses this feedback to update the models in online learning mode. When multiple passengers submit feedback for the same journey window, the Data Aggregator computes an aggregated MOS to reduce the impact of individual outliers. To protect passengers, TQMS does not persist raw ticket identifiers in the learning datastore. Instead, it stores a pseudonymous journey token computed as a salted, keyed hash (HMAC-SHA256) before any persistent storage, following the solution proposed in [33]. The Passenger Feedback Web-App uses explicit user consent: feedback is used for model adaptation only after the user actively agrees; otherwise, feedback is not used for online learning. To manage data retention, raw sensor streams are deleted after feature extraction and model update, while only aggregated features and versioned model snapshots are retained longer for monitoring and rollback.

Online learning exposes TQMS to poisoning attacks: a malicious actor may submit manipulated feedback to bias the neural network predictions. To mitigates this risk, TQMS adopts the solution in [34], which performs source evaluation and source filtering before aggregation. Within each update window, TQMS computes a consistency check between the submitted rating and the expected score inferred from the objective sensor features in the same time-aligned interval. Let $\hat{r}$ be the discrete quality level predicted by the current estimator from the objective features; TQMS marks a feedback r as suspicious and excludes it from MOS computation and online updates when $|r - \hat{r}| \geq 2$ on the 5-level scale. To reduce false rejections during highly dynamic driving segments, the threshold is relaxed when the window volatility is high (as measured from feature variance within the window).

V. EXPERIMENTAL EVALUATION

An experimental evaluation has been designed and conducted to assess the functionalities and performance of TQMS in a real-world public transport scenario. Sixty volunteers were recruited to perform an extensive evaluation of the driving style and transport quality for several bus lines of the public transport service in the city of Cagliari, which were equipped

with the proposed TQMS. The aim was to collect subjective quality ratings to be used as ground-truth quality scores for the objective quality measurements collected through the bus's on-board devices. Then, the performance of the driving style and transport quality prediction models, trained on the collected data, was computed.

A. SUBJECTIVE ASSESSMENT AND DATASET COLLECTION

Sixty volunteers (31 men and 29 women, age range 18–64 years, mean age 35.6 ± 12.4 years) participated in the experimental evaluation, each engaged in multiple bus rides on the public transport service in the city of Cagliari (Italy). In total, the participants completed 1,200 rides, with the range of rides per participant varying from 13 to 33. Participants were asked to leave each bus ride after about 10 minutes and evaluate the transport experience. All participants were fully informed about the purpose and procedures of the study and provided informed consent before participation. In particular, participants were instructed to evaluate two aspects: the perceived driving style and transport service quality during the ride. The former concerns the perceived ride comfort due to the driving behavior, whereas the latter regards the overall perceived service quality due to all journey aspects, including ride comfort, punctuality, cleanliness, etc. Upon completing their journey, participants used the Passenger Feedback Web-App to provide detailed feedback on these two metrics. The feedback was recorded using a Likert scale ranging from 1 to 5 (1-Bad, 2-Poor, 3-Sufficient, 4-Good, 5-Excellent). Participants sometimes provided qualitative comments about specific observations during the ride, such as road conditions and the occurrence of exceptional events (such as an incident). The assessment covered 20 bus lines of the local transport service provider, including urban, suburban, and mixed routes, with the number of rides per line ranging from 18 to 132. The inclusion of multiple lines allowed the study to assess the robustness of TQMS across different traffic conditions and passenger loads.

The Data Aggregator module associated the subjective feedback provided by the participants with the objective data collected by the on-board devices in the bus, based on the timestamps recorded for both objective and subjective evaluations, as explained in Section IV. Moreover, the Data Aggregator computed the statistical and temporal metrics in Table 2 on the collected sensor data.

B. INFLUENCE OF PARTICIPANTS' PROFILE

We analyzed the collected dataset to investigate whether the volunteer's profile had an influence on the perceived quality, with equal travelling conditions. At enrolment, each participant reported four attributes: age, gender, main purpose of bus travel, and frequency of bus use. Age was divided into three groups: 18–29, 30–44, and 45–64 years. Travel purpose was classified as work, study, or leisure, while frequency of use was classified as daily, about half of the days in a week, or occasional (one or two rides per week). Table 3 summarizes the composition of the participant sample.

TABLE 3. Four Self-Reported Attributes for the 60 Participants

Attribute	Level	Participants	Ride segments
Gender	Male	31	628
	Female	29	572
Age	18–29 years	22	443
	30–44 years	24	493
	45–64 years	14	264
Travel purpose	Work	27	559
	Study	18	361
	Leisure	15	280
Frequency of use	Daily	28	591
	About half the week	19	378
	Occasional (1–2 per week)	13	231

The Kruskal-Wallis test, with correction for tied ranks, was then applied to the ratings provided by the 60 volunteers during the evaluation, using each participant attribute as the grouping factor. For gender, which contains two groups, the Kruskal-Wallis test is equivalent to a rank-based two-group comparison. The test was applied separately to transport service quality and driving style. To account for the eight statistical comparisons, the resulting p-values were corrected using the Benjamini-Hochberg procedure, and the corresponding adjusted values are reported as q-values.

For transport service quality, no significant difference was observed between genders ($H(1) = 0.38$, $p = 0.538$, $q = 0.615$). Conversely, a significant effect was found for age ($H(2) = 9.71$, $p = 0.008$, $q = 0.029$) and travel purpose ($H(2) = 11.49$, $p = 0.003$, $q = 0.026$). Frequency of use showed a weaker trend that did not reach statistical significance after correction ($H(2) = 4.88$, $p = 0.087$, $q = 0.132$). For driving style, gender was again not significant ($H(1) = 0.17$, $p = 0.680$, $q = 0.680$). Age showed a significant effect ($H(2) = 9.06$, $p = 0.011$, $q = 0.029$), whereas neither travel purpose ($H(2) = 4.62$, $p = 0.099$, $q = 0.132$) nor frequency of use ($H(2) = 5.31$, $p = 0.070$, $q = 0.132$) was significant after correction.

Holm-corrected Dunn post-hoc comparisons showed that participants aged 18–29 years assigned lower ratings than participants aged 45–64 years for both transport service quality ($p_{\text{adj}} = 0.018$) and driving style ($p_{\text{adj}} = 0.024$). This result highlights that younger passengers tend to express more severe comfort judgements. For travel purpose, participants travelling mainly for work assigned lower service-quality ratings than those travelling mainly for leisure ($p_{\text{adj}} = 0.009$). No corresponding difference was found for driving style. This suggests that travel purpose mainly affects the evaluation of the overall service, including punctuality, crowding, and cleanliness, rather than the perception of vehicle dynamics alone. No other pairwise comparison was statistically significant.

Overall, the ratings were independent of gender and frequency of bus use, while age affected both driving style and transport service quality ratings, and travel purpose affected transport service quality ratings only. These effects do not prevent the use of the pooled dataset for training and evaluating

TABLE 4. Dataset Temporal Structure and Definition of the Temporal-Adjacency Window Used for Leakage Analysis

Property	Value
Total samples (ride segments)	1,200
Sampling rate (raw sensor log)	1 sample / 2 s
Average segment duration	10.0 ± 1.6 min
Sample granularity	1 row = 1 ride segment
Temporal-adjacency window (defined)	gap $\leq$ 10 min between consecutive ride segments
Same-subject & same-line pairs within window	174 consecutive pairs

the estimators, but they show that participant characteristics should be considered when interpreting the results and generalizing them to other passenger populations. Moreover, the subgroups differ in size, and the smallest group comprises 13 participants. The corresponding comparisons should therefore be interpreted with caution, by taking into account the size of the subgroups.

C. EVALUATION PROTOCOL AND FEATURE SELECTION

Table 4 summarizes the temporal structure of the dataset, which consists of 1,200 rides, each corresponding to a single ride segment of approximately 10 minutes, for which one feature vector is computed by the Data Aggregator. Among the consecutive ride-segment pairs, 174 correspond to cases in which the same participant continued travelling on the same bus line and vehicle from one segment to the next without alighting. These pairs are considered temporally adjacent and are used as the pairs in the temporal coherence term of (10). They represent genuine ride continuations for which encouraging coherence between consecutive quality estimates is physically meaningful.

To prevent data leakage across temporally-adjacent windows, we adopted exclusively group-aware partitioning protocols. We considered three complementary protocols: Leave-One-Line-Out (LOLO), which holds out all ride segments of one bus line to assess route transfer; Leave-One-Person-Out (LOPO), which holds out all ride segments of one participant to assess subject transfer; and a combined subject-and-route protocol, in which a group of participants and a group of bus lines are held out simultaneously, so that the test set contains only rides whose participant and bus line are both unseen during training. The Combined protocol is organized into 30 folds, obtained by partitioning the 60 participants into 6 groups and the 20 bus lines into 5 groups, and testing on the intersection of each participant group with each line group; the resulting test partitions contain between 25 and 66 rides. All three protocols guarantee, by construction, that every temporally-adjacent pair used by the temporal loss term falls entirely within either the training or the test partition, and never across the boundary.

Feature selection is performed independently within each cross-validation fold, using training data only, to prevent the risk of information leakage across temporally-adjacent windows. For each fold, the 396 candidate metrics of Table 2 are ranked according to their Kruskal-Wallis significance with respect to the corresponding subjective score, computed exclusively on the training partition of that fold. A metric was retained in a given fold when the test yielded $p < 0.001$; to account for multiple comparisons over the 396 candidate metrics, the Bonferroni-corrected threshold $p < 0.001/396 \approx 2.53 \times 10^{-6}$ was also applied. Across all 110 group-aware folds (60 LOPO, 20 LOLO, and 30 Combined), a stable subset of 30 features for transport service quality and 32 features for driving style was consistently selected in every fold, which are summarized by sensor category in Table 5.

TABLE 5. Features Consistently Selected Across All 110 Group-Aware Folds (60 LOPO, 20 LOLO, and 30 Combined), by Sensor Category, for Driving Style and Service Quality Estimation. Feature Selection Was Performed Independently Within Each Fold Using the Kruskal-Wallis Test Computed Exclusively on the Corresponding Training Partition.

Category	Original	Selected (all folds)	
		Service quality	Driving style
Acceleration	108	9	10
Angular velocity	54	7	8
Magnetic field	54	3	3
Euler angles	54	5	8
Quaternion orientation	72	2	0
Gravity vector	54	4	3
Total	**396**	**30**	**32**



FIGURE 6. Distribution of driving style and transport service quality scores.

D. PERFORMANCE OF PREDICTION MODELS

The resulting dataset showed class imbalance in the distributions of subjective driving style and transport service quality scores, as illustrated in Fig. 6. However, as explained in Section IV, the proposed one-vs-rest neural ensemble leverages a temporal coherence loss for service quality estimation,

TABLE 6. Comparison Between BCE and the Proposed Loss for Transport Service Quality Estimation Using LOLO

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.55	0.53	0.46	0.49	0.66	0.66	0.63	0.64
2	0.63	0.60	0.57	0.58	0.70	0.69	0.68	0.68
3	0.79	0.78	0.82	0.80	0.82	0.81	0.83	0.82
4	0.77	0.76	0.80	0.78	0.78	0.76	0.78	0.77
5	0.66	0.64	0.62	0.63	0.72	0.72	0.70	0.71
M-Avg	0.68	0.66	0.65	0.66	0.74	0.73	0.72	0.72
W-Avg	0.73	0.71	0.73	0.72	0.77	0.75	0.76	0.76

TABLE 7. Comparison Between BCE and the Proposed Loss for Driving Style Quality Estimation Using LOLO

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.52	0.48	0.44	0.46	0.63	0.63	0.59	0.61
2	0.60	0.56	0.53	0.54	0.68	0.68	0.65	0.66
3	0.67	0.64	0.63	0.63	0.73	0.73	0.72	0.72
4	0.79	0.78	0.82	0.80	0.82	0.83	0.81	0.82
5	0.81	0.80	0.84	0.82	0.81	0.81	0.82	0.81
M-Avg	0.68	0.65	0.65	0.65	0.73	0.74	0.72	0.72
W-Avg	0.71	0.69	0.70	0.70	0.76	0.76	0.75	0.75

TABLE 8. Comparison Between BCE and the Proposed Loss for Transport Service Quality Estimation Using LOPO

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.62	0.60	0.56	0.58	0.70	0.71	0.67	0.69
2	0.68	0.66	0.63	0.64	0.74	0.74	0.71	0.72
3	0.82	0.81	0.84	0.82	0.84	0.83	0.85	0.84
4	0.81	0.80	0.83	0.81	0.82	0.82	0.83	0.82
5	0.71	0.69	0.67	0.68	0.79	0.79	0.77	0.78
M-Avg	0.73	0.71	0.71	0.71	0.78	0.78	0.77	0.77
W-Avg	0.77	0.75	0.76	0.76	0.80	0.80	0.80	0.80

TABLE 9. Comparison Between BCE and the Proposed Loss for Driving Style Quality Estimation Using LOPO

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.57	0.54	0.50	0.52	0.67	0.67	0.64	0.65
2	0.63	0.60	0.56	0.58	0.71	0.70	0.68	0.69
3	0.70	0.68	0.64	0.66	0.75	0.74	0.72	0.73
4	0.81	0.80	0.82	0.81	0.83	0.82	0.83	0.82
5	0.83	0.82	0.84	0.83	0.82	0.81	0.81	0.81
M-Avg	0.71	0.69	0.67	0.68	0.76	0.75	0.74	0.74
W-Avg	0.74	0.72	0.72	0.72	0.78	0.77	0.76	0.76

TABLE 10. Comparison Between BCE and the Proposed Loss for Transport Service Quality Estimation Using the Combined Subject-and-Route Protocol

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.49	0.45	0.40	0.42	0.61	0.58	0.53	0.55
2	0.58	0.54	0.50	0.52	0.66	0.63	0.59	0.61
3	0.73	0.72	0.69	0.70	0.77	0.75	0.73	0.74
4	0.75	0.73	0.71	0.72	0.76	0.75	0.74	0.74
5	0.63	0.60	0.55	0.57	0.69	0.66	0.62	0.64
M-Avg	0.64	0.61	0.57	0.59	0.70	0.67	0.64	0.66
W-Avg	0.68	0.66	0.63	0.64	0.73	0.71	0.68	0.69

TABLE 11. Comparison Between BCE and the Proposed Loss for Driving Style Quality Estimation Using the Combined Subject-and-Route Protocol

	BCE				Proposed Loss			
Class	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
1	0.46	0.40	0.36	0.38	0.58	0.53	0.48	0.50
2	0.54	0.48	0.44	0.46	0.62	0.58	0.54	0.56
3	0.61	0.57	0.53	0.55	0.67	0.64	0.60	0.62
4	0.73	0.72	0.69	0.70	0.75	0.73	0.71	0.72
5	0.75	0.75	0.71	0.73	0.75	0.74	0.72	0.73
M-Avg	0.62	0.58	0.55	0.56	0.67	0.64	0.61	0.63
W-Avg	0.65	0.63	0.59	0.61	0.70	0.67	0.64	0.66

which allows the developed estimation models to balance the influence of each score class during training, enhancing generalization capability across imbalanced data distributions. To evaluate the contribution of the temporal coherence loss to the estimation performance, we have also trained the neural ensemble using the class-weighted BCE to compute baseline performance using a common loss function for training binary classification models. The proposed coherence temporal loss aims to add a temporal-coherence factor aimed at stabilizing mid-scale decisions without flattening genuine dynamics. This yields an ablation in which model, data, features and protocols are identical; only the loss changes.

To compute robust estimation performance, we considered three complementary group-aware evaluation protocols: LOLO to assess route transfer, LOPO to assess subject transfer, and the Combined subject-and-route protocol to assess simultaneous generalization to participants and bus lines that are both unseen during training. The metrics reported in Tables 6-11 were computed from the pooled out-of-fold predictions, so that each sample contributed to the test predictions exactly once. For the fold-level analysis, the macro-averaged F1-score was computed only over the classes with non-zero ground-truth support in each fold, and the same set of classes was used to evaluate all loss conditions. Therefore, a class absent from a test fold was not assigned an F1 value of zero, whereas a class present in the fold but never predicted received an F1 value of zero. Moreover, all tables report M-Avg and W-Avg, which are the macro-average and weighted-average, respectively.

As detailed in Section V-C, in all protocols the held-out partition is fixed before any data processing, feature selection and scaling are applied only to the training split of each fold, and the group structure (line, subject, or both) guarantees that no temporally-adjacent window pair is split between training and test. Both BCE and proposed-loss models are trained on identical folds with identical weight initialization within each fold, so that any difference in performance is attributable to the loss term itself.

Tables 6 and 7 summarize the transport service and driving style quality estimation performance, respectively, using the LOLO protocol and comparing BCE with the proposed loss function. The proposed loss increases weighted-average F1 over class-weighted BCE from 0.72 to 0.76 for service quality and from 0.70 to 0.75 for driving style, with the largest gains concentrated in the minority classes. The F1 for class 1 increases from 0.49 to 0.64 for service quality and from 0.46 to

0.61 for driving style. Conversely, service-quality class 4 decreases from 0.78 to 0.77, and driving-style class 5 decreases from 0.82 to 0.81.

Tables 8 and 9 summarize the transport service and driving style quality estimation performance, respectively, using the LOPO protocol and comparing BCE with the proposed loss function. Weighted-average F1 increases from 0.76 to 0.80 for service quality and from 0.72 to 0.76 for driving style, again driven primarily by gains on the minority classes. Driving-style class 5 is the only class showing a reduction under this protocol, from 0.83 to 0.81.

Tables 10 and 11 report the results of the combined subject-and-route protocol, which is the most demanding of the three since both the participant and the bus line of every test ride are unseen during training. As expected, the absolute performance is lower than under LOLO and LOPO, confirming that generalizing simultaneously across subjects and routes is harder than transferring along a single dimension. Nevertheless, the proposed loss improves the weighted-average F1 from 0.64 to 0.69 for service quality and from 0.61 to 0.66 for driving style. The corresponding macro-F1 values increase from 0.59 to 0.66 and from 0.56 to 0.63, respectively. This indicates that the benefit of the temporal coherence term is preserved when both dimensions are unseen during training.

We also observe that precision and recall for the minority classes improve together under the proposed loss rather than trading off against each other (e.g., class 1 in LOPO service quality: precision from 0.60 to 0.71 and recall from 0.56 to 0.67; class 1 in LOPO driving style: precision from 0.54 to 0.67 and recall from 0.50 to 0.64). This indicates that the temporal coherence term, combined with the inverse-frequency class weighting of (7a), stabilizes estimation for underrepresented quality levels without sacrificing sensitivity to them, while leaving performance on the majority classes largely unchanged. We further discuss this behavior, and its implications for classes with very few samples, in Section VI. Finally, due to space limitations, detailed ablation studies regarding the contribution of the variance-dependent gate compared with that of the temporal term are provided in the Supplementary Material.

VI. CHALLENGES AND LIMITATIONS

The proposed TQMS leverages IoT and AI-based technologies for monitoring and estimating driving style and service quality of public transport in a city. This is a challenging task because service quality depends on several aspects, such as the comfort provided by the bus (cleanliness, driving style, seat availability) and punctuality. In this research, we measured the driving style by collecting both objective and subjective data, and the service quality by collecting subjective data only. However, the proposed TQMS is scalable and open to further data sources, since the Bus Quality Model leverages APIs to receive data from external services: for instance, bus occupancy or cleanliness, obtained with computer vision algorithms analysing the images of bus security cameras, or bus punctuality, typically monitored by transport service providers. The Data Aggregator and Signal Quality Estimator modules can then be updated to treat more data and provide updated estimators.

Our study presents an empirical evaluation of TQMS using a participant pool of 60 individuals, balanced by gender (31 men and 29 women) and covering an age range of 18–64 years, over 1,200 bus rides on 20 different lines. While this sample allows the assessment of subject-, route-, and combined subject-and-route generalization, it still does not capture the full range of factors known to influence passenger-perceived quality, such as disability status, travel purpose, frequency of bus use, sensitivity to motion sickness, prior expectations, and cultural background. In particular, all participants were recruited in a single city and used the services of one transport operator, which limits the generalization of the reported perceptual outcomes to other geographical, operational, and cultural contexts.

Future work will address this limitation by extending the experimental campaign to a more heterogeneous population, varying age, disability status, travel habits, and geographical context. A further limitation concerns the least represented quality classes, on which the proposed loss yields its largest relative improvements. Indeed, quality score 1 is represented by only 78 service-quality and 112 driving-style samples. The corresponding estimates therefore remain subject to higher variance. Moreover, three of the 30 class-level F1 comparisons favor BCE, even though the aggregate improvement is preserved across all three protocols. A possible mitigation is to adopt a coarser scale, such as a 3-level or 2-level one, identifying "bad-sufficient-good" or "bad-good" quality values: while losing granularity, this would still allow detecting quality reductions, whose causes can be investigated through objective measurements.

VII. CONCLUSION

The proposed TQMS is an IoT solution devoted to the monitoring and prediction of bus transport service quality. It encompasses sensors, communication infrastructure, quality bus modeling, and predictive analytics, as well as quality visualization capabilities, and these features are design-ready for integration into DT solutions, an increasingly prevalent approach in transport service management. Driving style and service quality are estimated by deep learning-based estimators trained on both objective and subjective data; to reduce prediction jitter between consecutive estimation windows, due to small fluctuations in the bus sensor readings, the proposed one-vs-rest neural ensemble leverages a novel temporal coherence loss instead of a standard BCE loss. By associating orientation-related sensor data with subjective passenger feedback collected on urban buses of an Italian operator, TQMS achieved average macro-F1 scores of 0.72 for service quality and 0.70 for driving style across route-independent (LOLO), subject-independent (LOPO), and combined subject-and-route evaluation protocols. Moreover, the proposed loss improves macro-F1 by 0.06–0.07 and weighted-average F1 by 0.04–0.05, corresponding to an average weighted-F1 gain of

4.5 percentage points, with the largest gains concentrated on the least-represented quality classes. The improvement is preserved under the most demanding combined protocol, in which both the participant and the bus line of every test ride are unseen during training, although some individual classes favor BCE. Key contributions of this work include a scalable architecture adaptable to diverse public transport systems and an integrated framework bridging objective sensor measurements and subjective user evaluations, with edge-layer processing ensuring system responsiveness and cloud applications supporting centralized monitoring and decision-making for operators. Future work will focus on integrating further objective data contributing to the overall service quality and on extending the experimental campaign to a larger and more heterogeneous passenger population, so as to capture the full range of factors known to influence passenger-perceived quality in public bus transport.

REFERENCES

[1] A. Luhn, "2023 was officially the world's hottest year on record," *New Scientist*, vol. 261, no. 3473, pp. 9–9, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S026240792400054X?via%3Dihub

[2] P. Shukla et al., "Climate change 2022: Mitigation of climate change. cambridge," New York, NY, USA: Cambridge Univ. Press, 2022. [Online]. Available: https://www.ipcc.ch/report/ar6/wg3/downloads/report/IPCC_AR6_WGIII_Citation.pdf

[3] J. Göransson and H. Andersson, "Factors that make public transport systems attractive: A review of travel preferences and travel mode choices," *Eur. Transp. Res. Rev.*, vol. 15, no. 1, Sep. 2023, Art. no. 32.

[4] B. Barabino, N. Cabras, C. Conversano, and A. Olivo, "An integrated approach to select key quality indicators in transit services," *Social Indicators Res.*, vol. 149, pp. 1045–1080, 2020.

[5] B. Barabino and M. Di Francesco, "Characterizing, measuring, and managing transit service quality," *J. Adv. Transp.*, vol. 50, no. 5, pp. 818–840, 2016.

[6] T. Nguyen, D. Nguyen-Phuoc, and Y. Wong, "Developing artificial neural networks to estimate real-time onboard bus ride comfort," *Neural Comput. Appl.*, vol. 33, pp. 5287–5299, 2021.

[7] B. Barabino, M. Coni, A. Olivo, G. Pungillo, and N. Rassu, "Standing passenger comfort: A new scale for evaluating the real-time driving style of bus transit services," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4665–4678, Dec. 2019.

[8] J. Wang, J. Fan, M. Xie, and L. Cai, "Can variations of vehicle driving status provide accurate predictors of discomfort? A study on the actual driving test," *IEEE Access*, vol. 9, pp. 47024–47032, 2021.

[9] M. Coni et al., "On-board comfort of different age passengers and bus-lane characteristics," in *Proc. Comput. Sci. Appl.*, 2020, pp. 658–672.

[10] B. Manandhar, K. D. Vance, D. B. Rawat, and N. Yilmaz, "Leveraging digital twin technology for sustainable and efficient public transportation," *Appl. Sci.*, vol. 15, no. 6, 2025, Art. no. 2942.

[11] M. Andruccioli, G. Delnevo, R. Girau, and P. Salomoni, "Digital twins in public bus transport: A systematic literature review of architectures, intelligence, and interaction," *Future Gener. Comput. Syst.*, vol. 180, 2026, Art. no. 108379.

[12] N. Moreno, L. Toro-Gálvez, J. Troya, and C. Canal, "Modeling urban digital twins over the cloud-to-thing continuum," in *Proc. MESS'23: 3rd Int. Workshop MDE Smart IoT Syst.*, 2023, pp. 1–5.

[13] P. Ruiz, M. Seredynski, Á. Torné, and B. Dorronsoro, "A digital twin for bus operation in public urban transportation systems," in *Proc. Big Data Intell. Comput.*, in Lecture Notes in Computer Science, 2023, pp. 40–52.

[14] S. Deng, J. Zhong, S. Chen, and Z. He, "Digital twin modeling for demand responsive transit," in *Proc. 1st IEEE Int. Conf. Digit. Twins Parallel Intell.*, 2021, pp. 410–413.

[15] O. Yusuf, A. Rasheed, F. Lindseth, and M. Slaastuen, "Unveiling urban mobility patterns: A data-driven analysis of public transit," in *Proc. Int. Conf. Control, Autom. Diagnosis*, 2024, pp. 1–6.

[16] A. Amrani et al., "Architecture of a public transport supervision system using hybridization models based on real and predictive data," in *Proc. 23rd Euromicro Conf. Digit. System Des.*, 2020, pp. 440–446.

[17] V. K. Verma and R. Rastogi, "How do stakeholders perceive transit service quality attributes? A study through Fuzzy-AHP," *Expert Syst. Appl.*, vol. 238, 2024, Art. no. 122043.

[18] M. A. Saleem, H. Afzal, F. Ahmad, H. Ismail, and N. Nguyen, "An exploration and importance-performance analysis of bus rapid transit systems' service quality attributes: Evidence from an emerging economy," *Transport Policy*, vol. 141, pp. 1–13, 2023.

[19] F. Maltinti, M. Coni, J. Rombi, B. Barabino, R. Ventura, and N. Rassu, "A literature review on bus comfort on-board," in *Proc. Comput. Sci. Appl. Workshops*, 2024, pp. 80–97.

[20] F. Maltinti et al., "Vulnerable users and public transport service: Analysis on expected and perceived quality data," in *Proc. Comput. Sci. Appl.*, 2020, pp. 673–689.

[21] S. Deb, M. Ali Ahmed, and D. Das, "Service quality estimation and improvement plan of bus service: A perception and expectation based analysis," *Case Stud. Transport Policy*, vol. 10, no. 3, pp. 1775–1789, 2022.

[22] D. Sekulić, S. Rusov, V. Dedović, S. Šalinić, D. Mladenović, and I. Ivković, "Analysis of bus users' vibration exposure time," *Int. J. Ind. Ergonom.*, vol. 65, pp. 26–35, 2018.

[23] G. Wang, J. Zhang, and X. Kong, "Study on passenger comfort based on human–bus–road coupled vibration," *Appl. Sci.*, vol. 10, no. 9, 2020, Art. no. 3254.

[24] J. Conde, A. Munoz-Arcentales, A. Alonso, S. Lopez-Pernas, and J. Salvachua, "Modeling digital twin data and architecture: A building guide with FIWARE as enabling technology," *IEEE Internet Comput.*, vol. 26, no. 3, pp. 7–14, May/Jun. 2022.

[25] A. De Benedictis, F. Rocco di Torrepadula, and A. Somma, "A digital twin architecture for intelligent public transportation systems: A FIWARE-Based solution," in *Proc. the 21st Int. Symp. Web Wireless Geographical Inf. Syst.*, pp. 165–182.

[26] Y. Gao, S. Qian, Z. Li, P. Wang, F. Wang, and Q. He, "Digital twin and its application in transportation infrastructure," in *Proc. IEEE 1st Int. Conf. Digit. Twins Parallel Intell.*, 2021, pp. 298–301.

[27] O. E. Marai, S. Messinis, N. Doulamis, T. Taleb, and J. Manner, "Roads infrastructure digital twin: Advancing situational awareness through bandwidth-aware 360° video streaming and multi-view clustering," *IEEE Open J. Veh. Technol.*, vol. 6, pp. 1348–1362, 2025.

[28] C. Rose, T. Ahmad, R. McMurray, T. R. Gadekallu, and M. U. Hadi, "Digital twin enabled autonomous vehicles for obstacle avoidance in disaster response," *IEEE Open J. Veh. Technol.*, vol. 7, pp. 1755–1769, 2026.

[29] K. Duran, L. V. Cakir, A. Fonzone, T. Q. Duong, and B. Canberk, "Digital twin-empowered green mobility management in next-gen transportation networks," *IEEE Open J. Veh. Technol.*, vol. 5, pp. 1650–1662, 2024.

[30] W. A. Ali, M. Roccotelli, and M. P. Fanti, "Digital twin in intelligent transportation systems: A review," in *Proc. 8th Int. Conf. Control, Decis. Inf. Techn.*, 2022, vol. 1, pp. 576–581.

[31] M. B. Ribeiro et al., "A digital twin simulation-based framework to the public transportation network in lisbon," *IFAC-PapersOnLine*, vol. 59, no. 10, pp. 2433–2438, 2025.

[32] X. Gu et al., "Digital twin technology for intelligent vehicles and transportation systems: A survey on applications, challenges and future directions," *IEEE Commun. Surveys Tuts.*, vol. 28, pp. 3235–3271, 2026.

[33] I. Dia, B. Sy, O. Diagne, S. Mané, and L. Diouf, "Real-time quarry truck monitoring with deep learning and license plate recognition: Weighbridge reconciliation for production control," *Mining*, vol. 5, no. 4, 2025, Art. no. 84.

[34] Z. Huang, M. Pan, and Y. Gong, "Robust truth discovery against data poisoning in mobile crowdsensing," in *Proc. IEEE Glob. Commun. Conf.*, 2019, pp. 1–6.