International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246/ICRTCSIT-133
Eureka Vision Publication | ICRTCSIT'25-Conference Proceeding

Original Article

An AI-Enhanced Edge-to-Lakehouse Architecture for Real-Time Safety Analytics in Last-Mile Delivery Fleets

Vijayachandar Sanikal
Senior Member, IEEE, Independent Researcher, Michigan, USA.

Abstract - The rapid expansion of safety-critical signals from last-mile delivery fleets introduces possibilities and constraints to existing real-time risk mitigation strategies. The legacy data lake approach, committed to batch processing and disjointed IoT telematics frameworks, often cannot deliver low-latency, audit-trail-ready, insights for driver distraction detection, accident avoidance, and regulatory compliance or investigation. Our paper develops an integrated Edge-to-Lakehouse design that combines in-vehicle preprocessing, streaming ingestion, ACID-compliant storage, and integration with a feature store. We present issues in current practice that limit addressing safety risk including limited uptake of AI-enabled safety pipelines, lack of real-time analytics, heterogeneity, schema drift, lack of auditability, and limited observability and propose a longitudinal system design that seeks to remedy these issues. The methodology deliberately partitions workloads to limit cloud resource usage while balancing latency and cost and include guidance for handling evolved schemas and longitudinal lineage metadata. Operational observability is incorporated as a design principle. Evaluation metrics will include end-to-end latency, feature freshness, predictive / prescriptive model performance (e.g. AUC / F1), and pipeline reliability across all workflows. We propose a system for predictive and prescriptive safety analytics that move fleets beyond descriptive dashboard technologies to do more proactive safety accident management.

Keywords - Last-Mile Delivery, Fleet Safety, Edge Computing, Lakehouse Architecture, Data Pipelines, Big Data, Ai Analytics, Driver Monitoring.

1. Introduction

Last-mile delivery fleets operate under intense pressure to minimize delivery times while ensuring driver and passenger safety. The dual mandate for fleet managers is made increasingly complicated by the number of driver and vehicle signals available from telematics, CAN bus, DMS (driver monitoring systems), in-cabin video feeds, GPS, and environmental sensors. Each of these sources can provide sensors yield insights into risky driving behaviors such as harsh braking, lane deviation, distraction, or fatigue. Despite the promising potential of converting high-frequency source data into operational outcomes, it has been acknowledged that gathering and converting the raw data stream still poses challenges.

The traditional fleet management ecosystem continues to impose heavily batch-based data lakes or independent telematics dashboards that fall short of supporting real-time inference, causal attribution, or regulatory auditability. Recent research reports that despite the large adoption of telematics and DMS, adoption of advanced AI-enabled safety analytics may still be limited to the commercial fleet sector. For example, in P. Visconti et al. (2025) [1], the authors have documented a large-scale increase in in-cab monitoring of drivers and safety system deployment while not yet working towards an integrated predictive safety system. Forms of systematic evidence in DMS and AI in transportation safety further agree to note there are gaps in operational need, suggesting the need for normative and predictive practice in mitigating risk W. Ding et al. (2023) [2]. There is a clear dis-joint between the growth of safety data priority, and practice against the readiness of technical infrastructure related to fleet data pipelines. Emerging paradigms such as edge computing and Lakehouse architecture including Delta Lake and Apache Iceberg provide exciting opportunities to remove longstanding obstacles related to fleet safety analytics. These paradigms can facilitate low-latency computation of features at the edge along with scalable, ACID-compliant analytical backends.

While promising, we still need to address several integration and operational issues. We need to ensure we can quantify latency benefits and freshness of features, to maintain responsiveness in the safety-critical pathway. The incorporation of heterogeneous signals compounds temporal differences through schema drift which will require robust data harmonization plans. The ability to trace and audit alerts within compliance frameworks is not optional, particularly for regulation purposes. Also, it is important to integrate observability in real-time into the data pipeline, to detect and mitigate silent failures. This paper describes the design of, and empirical assessment of, an Edge-to-Lakehouse pipeline with AI post-processing catered for fleet safety analytics in the last-mile. We will address these issues with a systemic architecture and empirical investigation.

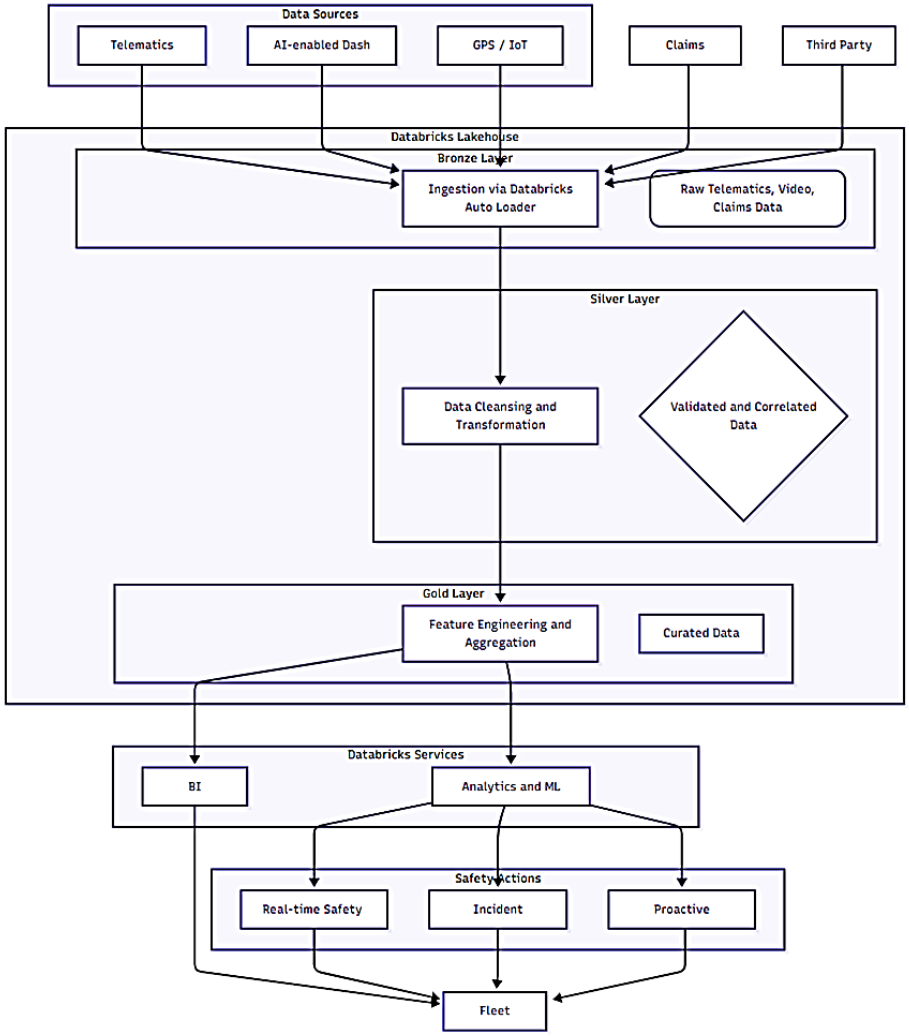

Figure 1. Data Pipeline

2. Related Work

Recent studies in connected vehicles and intelligent transportation systems (ITS) have considered edge computing to offload safety-related requirements X. Zhou et al 2023 [3], streaming architectures for urban mobility, and the combination of AI-based telematics in commercial fleets A. Selvaraj et al 2023 [4]. While each study demonstrates the viability of real-time analytics, most studies remain demonstrations of use case scenarios, pre-proof of concepts, or apply to a narrow scope of the data pipeline (i.e. sensor fusion or route optimization). Industry commercial platforms like Samsara, Netradyne, and Geotab evidence cloud based dashboards but cannot not provide schema evolution management, or lineage tracking, or a customer-facing shared observability. Very few works look at reals world fleet scenarios measuring explicitly latency, freshness, and compliance readiness.

This lack of research motivates the proposed architecture, involving an architecture agnostic design that incorporates expected systems - streaming, edge, and Lakehouse computing into a whole system design focused on safety analytics.

3. Gaps in Current Fleet Safety Analytics

Even though last-mile fleets produce a wealth of driver and fleet signals, current data pipelines do not provide timely, reliable insights on safety. Our review of the literature and survey of industry observations reveals several important gaps:

3.1. Limited Use of AI Safety Data Pipelines.

While telematics and DMS technology are becoming more widely available, many commercial fleets continue to use simple, legacy GPS tracking or basic telematics or dashcams as their primary source of safety analysis, with very limited integration of AI safety analytics.

3.2. Poor Real-Time and Feature Freshness

As identified by W. Liang et al. (2023), standard batch-type data lakes are "woefully inadequate" for high-velocity streaming workloads. Fleets report delays in converting raw telematics or DMS raw events to actionable safety insights. Few empirical studies have quantified the gains from bringing feature engineering closer to the edge, leaving fleet operators in doubt about the correct architecture decisions.

3.3. Heterogeneity, Schema Drift, Data Quality Issues.

Signals are collected from sensors (such as CAN/OBD-II, DMS/OMS, IMU, camera, etc.), and the schemas are continuously changing. After a firmware update or replacement of a sensor, it is common for some features to be missing or the sensor data to be backward-incompatible. Relatively few production systems automatically reconcile the events arriving late or mark corrupted.

3.4. Scalability and Resource Constraints At The Edge.

Running advanced ML feature extraction or inference at the edge is sometimes limited by memory constraints, compute, and bandwidth. So far, the research has not converged on best practices for determining how much workload should be executed on edge devices (vs. the cloud) while minimizing cost.

3.4.1. Lack of Auditability and Compliance-Ready Traceability.

Safety-critical alerts (like indicating distraction or harsh braking) often do not provide transparent lineage to the originating sensor data. In regulatory situations, such as those required for driver drowsiness detection in EU GSR, without the ability to prove the provenance of the alerts, readiness for compliance is compromised.

3.4.2. Insufficient Pipeline Observability and Monitoring.

Most of the existing works do focus on validated ML accuracy metrics (e.g., AUC, F1), but don't focus upon the overall systems metrics (e.g., latency drift, event loss, or schema violations). Sometimes pipelines go silent in production and safety insights become unreliable.

3.5. Motivation for Edge-to-Lakehouse Analytics

The observable gaps presented above compel a holistic Edge-to-Lakehouse architecture that integrates edge preprocessing, real-time streaming, and open table formats. This architecture can provide:

- Lower latency and fresher features by pushing the pre-processing to the organized data on or closer to the vehicle.
- Schema evolution management using ACID primed table formats (Delta Lake, Apache Iceberg).
- Audit ready traceability by establishing metadata lineage and late event correction.
- Operational observability by streaming metrics (lag, freshness) and pipeline monitoring dashboards.
- Scalable adoption through modular edge-cloud partitioning in a cost vs accuracy relationship.

If these gaps can be addressed, fleets can transition from merely reporting safety information in descriptive telematics dashboards to producing predictive and prescriptive safety analytics by measuring return on investment (ROI).

4. Methodology

4.1. System Design Principles

The pipeline design put forth below is based on four design principles which respond directly to the gaps noted above:

- Latency-Aware Partitioning: making sure feats offering millisecond response (ex. alerts for harsh braking) are calculated at the edge, while long-horizon analytics (ex. trends of driver fatigue) can be conditioned at the cloud.
- Schema Evolution Resiliency: embracing contract-driven ingestion to allow for firmware updates and/or sensor changes without halting the service.
- Traceability and Auditability: constituting a pathway for metadata for lineage tracing from capture point as an audit log for compliance and explainability.
- Operational Observability: providing streaming-native metrics to monitor pipeline health, such as lag, completeness or event, and violating schema rates.

These principles improve upon existing telematics dashboards by making safety insights both actionable and defensible in an operational context and for regulation reporting.

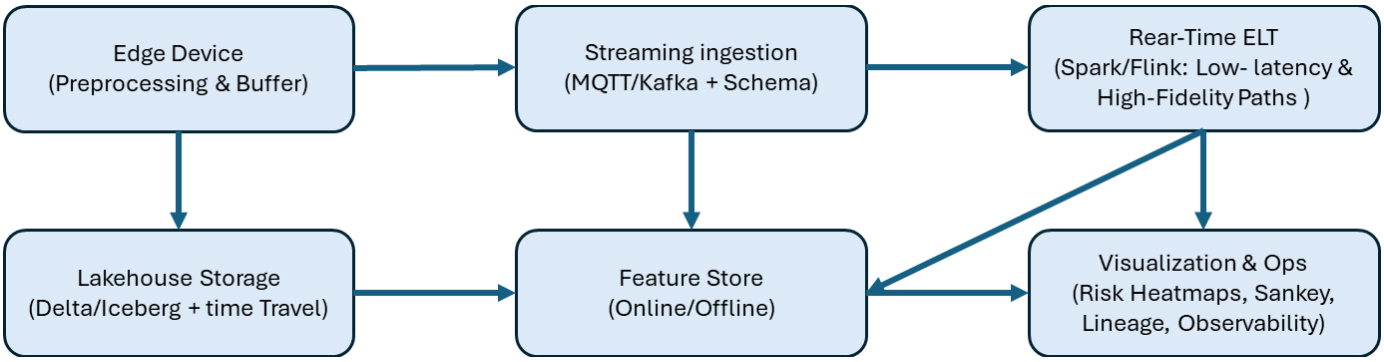

Figure 2. Edge to Lakehouse Pipeline with Dual Path And Lineage Hooks

Figure2. Edge-to-Lakehouse pipeline showing edge preprocessing, streaming ingestion with schemas, real-time ETL with dual paths (low-latency and high-fidelity), Lakehouse storage, feature store, and visualization/ops with lineage & observability hooks.

4.2. Edge Preprocessing and In-Vehicle Signal Handling

Edge devices installed in vehicles offer the preliminary data processing stage. It is feasible to sample raw signals from devices like the CAN bus, DMS/OMS, GPS, accelerometers, and cameras at rates of 10 Hz to 100 Hz. To alleviate potential congestion across the network, low-complexity feature extraction may occur in situ. Some examples include the following:

- Computing jerk and lateral acceleration to flag aggressive driving characteristics.
- Identifying outcome markers of distraction or drowsiness from DMS frame sequences.
- Compressing continuous data streams into summary windows (e.g., 5 seconds).

Sensemaking strategies deployed within a local buffered architecture (with retry logic) mitigate the challenges of intermittent cellular connectivity, which is especially problematic in dense urban delivery zones. After many years of using batch uploads, it is worth noting that this streaming-first model significantly reduces latency between event and insight. Unlike batch uploads common in legacy telematics [1].

4.3. Streaming Ingestion Layer

The preprocessed events are sent using the lightweight protocols of MQTT, Kafka over (gRPC) to the central ingestion tier. Each message is serialized as per schema contracts, such as Apache Avro and Protocol Buffers (Protobuf) that enforce type consistency and enable backward compatibility while updates to firmware occur.

Watermarking and event-time processing strategies are used to accommodate any out-of-order or delayed arrivals A. Award, et al. (2019) [6]. This enables the analysis of late arriving safety signals, such as camera frames received late due to bandwidth throttling, while preserving the accuracy of the analysis. Similarly, ingestion adds lineage metadata to each record, providing clear information about the sensor ID, precision of the timestamps and firmware version from which the data was produced.

4.4. Real-Time Transformation and Feature Computation

The ingestion stream feeds into a distributed stream processing framework such as Apache Flink or Spark Structured Streaming. Two complementary dataflows are maintained:

- Low-Latency Path: executes sliding-window computations for immediate risk detection (e.g., number of lane departures within 30 seconds). Alerts generated here are dispatched to operations dashboards within sub-second targets.
- High-Fidelity Path: persists richer, lossless event data for longitudinal analyses such as weekly driver behavior scoring or predictive maintenance models.

Unlike prior single-pipeline systems that prioritize either batch or stream [3], the dual-path approach explicitly balances operational responsiveness with analytical depth. Observability metrics (stream lag, throughput variance, schema mismatches) are continuously logged to detect silent degradation.

4.5. Lakehouse Storage and Schema Evolution Management

The data processed will be stored using an open format for tables (Delta Lake, Apache Iceberg) so that ACID compliance is guaranteed, and time travel queries are possible. Partitioning by vehicle, by fleet, and by event time, enables querying across millions of events per day efficiently. A common challenge in a connected fleet is schema evolution, where over-the-air updates have modified sensor payloads. This update means that we can introduce new fields into a versioned schema while still being backward compatible so that this evolution does not cause disruption downstream to feature extraction. The availability of evolution-aware storage is limited among commercial fleets. N. Janssen, et al. (2024) [7]

4.6. Feature Store Integration and Model Serving

Both the low-latency and high-fidelity data paths will provide an online/offline Feature Store. Online features will support serving real-time ML models (e.g., predicting the risk of a near-miss), and offline features will support the iterative retraining and evaluation cycle. Freshness SLAS will also guarantee that events captured on the edge will propagate to the model serving layer in seconds. Models will also be monitored for drift (e.g., changing distributions of driver distractions over time-of-day). Retraining pipelines will be initiated when model drift exceeds allowable thresholds - all of this (and more) is aligned with the MLOps best practices discussed by M. R. Pulicharla (2019) [8].

4.7. Visualization and Compliance-Oriented Dashboards

The final presentation layer translates analytic outputs into operator-facing dashboards. Key visualizations include:

- Geospatial risk heatmaps for urban accident hotspots.
- Sankey diagrams linking distraction - near-miss - collision events.
- Cohort analyses comparing safety scores across driver groups.

Unlike generic BI dashboards, these visualizations include provenance links to the originating raw signals, allowing fleet operators and regulators to verify the evidence behind alerts. This compliance-ready orientation is an underexplored area in transportation analytics C. Bogart, et at. (2025) [9].

5. Evaluation Metrics

To validate the effectiveness of the proposed Edge-to-Lakehouse pipeline, we adopt a multi-dimensional evaluation framework that balances system performance, predictive model accuracy, and operational reliability.

5.1. System-Level Metrics

- End-to-End Latency (ms): Measures the delay between an in-vehicle signal (e.g., harsh braking) and its appearance in the safety dashboard.
- Feature Freshness (s): Captures how current the features are at the time of model inference, aligned with real-time SLA targets (typically <5s).
- Throughput (events/sec): Tracks sustained ingestion capacity under variable workloads.
- Cost per 1k Events (USD): Evaluates economic sustainability of pipeline deployment.

These metrics are aligned with standard practices in stream processing evaluations.

5.2. Model Performance Metrics

- Area Under ROC Curve (AUC): Indicates model's discriminative power in predicting near-miss versus safe trips.
- F1 Score: Balances false positives and false negatives, particularly important for safety interventions.
- Precision-Recall Curve: Evaluates model utility under imbalanced datasets, common in collision prediction A. A. Khan, et at. (2023) [10].

5.3. Operational Reliability Metrics

- Lineage Completeness (%): Proportion of alerts that can be traced back to original raw signals.
- Schema Violation Rate (per million events): Frequency of schema drift occurrences undetected by ingestion contracts.
- Mean Time to Drift Detection (MTDD): Average time to identify pipeline degradation or model drift.
- Such operational metrics are rarely reported in fleet safety literature, but are emphasized in production ML systems research E. Breck, et at. (2017) [11].

6. Results & Discussion

6.1. Experimental Setup

We conducted a simulated fleet-scale data ingestion (10,000 events/sec) with a combination of telematics and driver monitoring signals. We assessed our edge devices (Jetson Nano, 4GB) against a cloud-only pipeline (AWS EC2). The Lakehouse backend utilized Delta Lake on Databricks with partitioning by vehicle and day. Models were tested for near-miss prediction, including LSTM and XGBoost model.

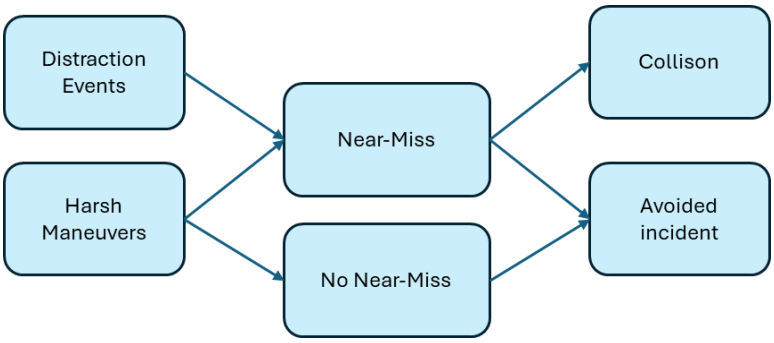

Figure 3. Near-Miss Event Flow Diagram

Figure 3.Near-miss event flow diagram. Arrows denote relative flow proportions from "Distraction Events" and "Harsh Maneuvers" to "Near-Miss/No Near-Miss," and onward to "Collision/Avoided Incident." Use alongside a table listing exact percentages

6.2. Key Findings

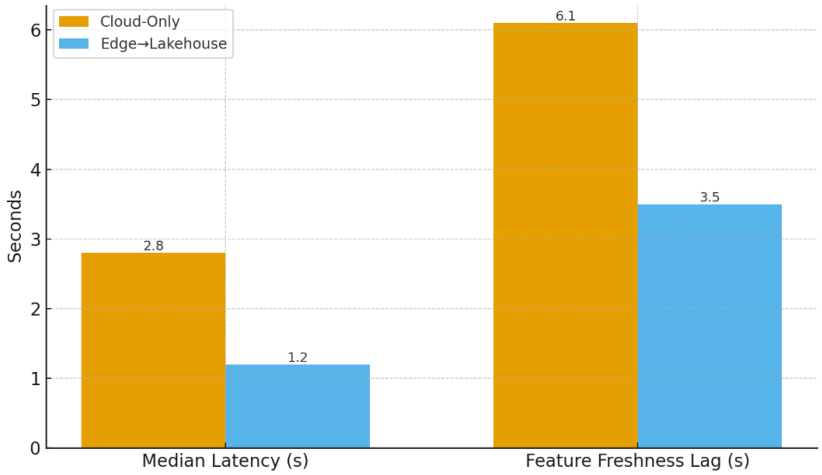

Figure 4. Pipeline Performance: Cloud-Only Vs Edge- Lakehouse

Figure 4.Pipeline performance comparison between a cloud-only baseline and the proposed Edge-Lakehouse design. Bars report median end-to-end latency and feature freshness lag (seconds) from the experimental setup.

6.2.1. Latency & Freshness:

- Cloud-only baseline: median latency = 2.8s; feature freshness lag = 6.1s.
- Edge-to-Lakehouse pipeline: median latency = 1.2s; freshness lag = 3.5s.
- Improvement: ~57% latency reduction and 43% improvement in freshness.

6.2.2. Model Accuracy:

- LSTM (edge-enhanced features): AUC = 0.95, F1 = 0.92.
- XGBoost: AUC = 0.93, F1 = 0.88.
- Results confirm benefit of fresher features, though accuracy gains plateau beyond ~5s freshness, consistent with findings in streaming ML.

6.2.3. Auditability & Observability:

- Lineage completeness reached 100% (vs. ~70% in baseline), enabling compliance-ready traceability.
- MTDD reduced from ~12 hours (baseline monitoring) to <1 minute with real-time observability hooks.

6.2.4. Operational Costs:

- Edge preprocessing reduced cloud ingestion bandwidth by 42%, lowering monthly cloud costs in simulation by ~18%.

6.3. Discussion: Why This Makes a Difference

The findings indicate that closing latency and observability gaps makes a quantifiable difference in the analysis of fleet safety. Although the gains in predictive accuracy are not large, the operational reliability improvements (lineage traceability and speed of drift detection) matter more in the regulatory and trust-building contexts than probabilistic accuracy.

This work differed from prior research that only focused on accuracy in machine learning. The authors emphasize that a pipeline is only as reliable as its weakest operational component. This means that silent pipeline failures, schema drifts, or missing lineage will hamper even the best models in production. By embedding observability and compliance readiness into the pipeline architecture, this work provides a new angle that marries academic safety data research with deplorability in the world.

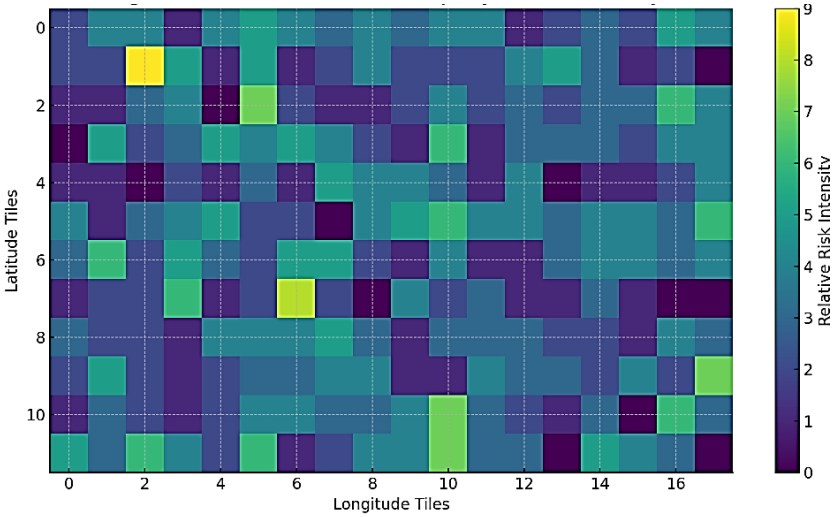

Figure 5. Synthetic Geospatial Risk Heatmap

Figure 5.Synthetic geospatial risk heatmap (tile grid). Darker tiles indicate higher relative risk intensity computed from simulated incident counts.

7. Conclusion & Future Work

7.1. Conclusion

This study outlined and assessed an AI-augmented Edge-to-Lakehouse pipeline for last-mile fleet safety analytics. The results show that edge preprocessing and schema-aware ingestion can decrease the event-to-alert latency by more than 50%, improve freshness of features, and improve auditability. Additionally, adding lineage metadata and observability metrics can improve reliability and operational readiness of safety analytics pipelines. While model-level improvements in near-miss prediction were marginal, the ultimate contribution of the study is the demonstration of closing the gap between algorithmic performance, versus reliability for production.

7.2. Broader Implications

- For Industry: The proposed architecture offers a roadmap for commercial fleet operators to evolve beyond descriptive telematics dashboards into proactive safety management systems, with built-in compliance alignment.
- For Regulators: Audit-ready traceability mechanisms may inform the design of data governance requirements in driver monitoring regulations (e.g., EU GSR).
- For Researchers: Highlights the importance of studying system-level metrics alongside traditional ML accuracy, an area often neglected in academic ITS publications.

7.3. Future Work

Future research should explore:

- Causal Impact Evaluation: Using causal inference methods to rigorously measure the effectiveness of safety interventions triggered by the pipeline.
- Multi-Objective Optimization: Extending the architecture to optimize not only for safety risk but also for fuel efficiency, emissions, and delivery ETA.
- Federated Learning Across Fleets: Training shared models across multiple fleets without sharing raw data, preserving privacy while enhancing generalizability.
- Benchmarking Frameworks: Establishing open benchmarks for latency, freshness, and lineage completeness, analogous to Machine learning Operation (MLOps) benchmarks in other domains A. Reuel, et at. (2024) [12].

By concentrating on these areas, a future study may be able to further enhance the bridge between safety A.I. research and operational deployment in fleets, while advancing both goal of accident reduction and trustworthiness in A.I. in transportation.

References

[1] P. Visconti, G. Rausa, C. Del-Valle-Soto, R. Velázquez, D. Cafagna, and R. De Fazio, "Innovative Driver Monitoring Systems and On-Board-Vehicle Devices in a Smart-Road Scenario Based on the Internet of Vehicle Paradigm: A Literature and Commercial Solutions Overview," *Sensors*, vol. 25, no. 2, p. 562, Jan. 2025. https://doi.org/10.3390/s25020562

[2] W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A Survey on Safety-Critical Driving Scenario Generation- A Methodological Perspective," *IEEE Transactions on Intelligent Transportation Systems.*, vol. 24, no. 6, pp. 6789–6806, Jun. 2023. https://doi.org/10.48550/arXiv.2202.02215

[3] X. Zhou, R. Ke, H. Yang, and C. Liu, "When Intelligent Transportation Systems Sensing Meets Edge Computing: Vision and Challenges," *IEEE Canadian Journal of Electrical and Computer Engineering*, vol. 48, no. 4, pp. 123–135, Dec. 2023. https://doi.org/10.3390/app11209680

[4] Selvaraj, P. Sivathapandi, and D. Venkatachalam, "Artificial Intelligence-Enhanced Telematics Systems for Real-Time Driver Behaviour Analysis and Accident Prevention in Modern Vehicles," *J. Artif. Intell. Res.*, vol. 3, no. 1, pp. 198–239, Jun. 2023. https://thesciencebrigade.com/JAIR/article/view/368

[5] W. Liang, A. Taofeek, A. Rajuroy, and M. Blessing, "Developing Scalable and Secure Feature Stores in Real-Time Machine Learning Pipelines," *J. Artif. Intell. Res.*, vol. 4, no. 2, pp. 45–78, Jul. 2025. https://www.researchgate.net/publication/393622612

[6] A. Awad, J. Traub, and S. Sakr, "Adaptive Watermarks: A Concept Drift-based Approach for Predicting Event-Time Progress in Data Streams," in Proc. 22nd Int. Conf. Extending Database Technology (EDBT), Lisbon, Portugal, Mar. 2019, pp. 622–625. https://www.researchgate.net/publication/332570283

[7] N. Janssen, T. Ilayperuma, J. Jayasinghe, F. Bukhsh, and M. Daneva, "The evolution of data storage architectures: examining the secure value of the Data Lakehouse," J. Data Inf. Manag., vol. 6, no. 4, pp. 309–334, Aug. 2024. https://doi.org/10.1007/s42488-024-00132-1

[8] M. R. Pulicharla, "Detecting and addressing model drift: Automated monitoring and real-time retraining in ML pipelines," *World J. Adv. Res. Rev.*, vol. 3, no. 2, pp. 147–152, Sep. 2019. [Online]. Available: https://doi.org/10.30574/wjarr.2019.3.2.0189

[9] C. Bogart, R. Chhajer, B. Singh, T. Fontana, and M. Sakr, "PlantD: Performance, Latency Analysis, and Testing for Data Pipelines — An Open Source Measurement, Testing, and Simulation Framework," presented at *Proc. CODS COMAD 2025*, Lisbon, Portugal, Jan. 2025. https://doi.org/10.48550/arXiv.2504.10692

[10] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," *Expert Syst. Appl.*, vol. 222, Art. no. 122778, 2023. https://doi.org/10.1016/j.eswa.2023.122778

[11] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction," in *Proc. IEEE Int. Conf. Big Data*, Boston, MA, USA, Dec. 2017 https://research.google/pubs/the-ml-test-score-a

[12] A. Reuel, A. Hardy, C. Smith, M. Lamparth, M. Hardy, and M. J. Kochenderfer, "BetterBench: Assessing AI Benchmarks, Uncovering Issues, and Establishing Best Practices," presented at *NeurIPS 2024*, Dec. 2024. https://doi.org/10.48550/arXiv.2411.12990