



OPEN ACCESS

Volume: 5

Issue: 4

Month: October

Year: 2026

ISSN: 2583-7117

Published: 01.10.2026

Citation:

Sai Rohit Jonnada, Jaya Surya Teja Gaddam, Mohan Krishna Oorugonda, "LLM-Powered Root Cause Intelligence for Inventory and Shipment Discrepancies in Warehouse Management Systems" International Journal of Innovations in Science Engineering and Management, vol. 5, no. 4, 2026, pp. 30-38.

DOI:

10.69968/ijisem.2026v5i430-38



This work is licensed under a Creative Commons Attribution-Share Alike 4.0 International License

LLM-Powered Root Cause Intelligence for Inventory and Shipment Discrepancies in Warehouse Management Systems

Sai Rohit Jonnada¹, Jaya Surya Teja Gaddam², Mohan Krishna Oorugonda³

¹Louisville, Kentucky, USA, 40222, sairohitjonnada97@gmail.com

²Louisville, Kentucky, USA, 4024, jsuryatejagaddam@gmail.com

³Coppell, Texas, USA, 75019, oorugondamohan@gmail.com

Abstract

While there are many differences between inventory and shipments in Warehouse Management Systems (WMS), the most serious ones come with a high price tag in global supply chains; and current diagnostic approaches are fragmented, reactive, and too much dependent on domain-expert intervention. The authors propose the LLM-Powered Root Cause Intelligence (LRCI) framework, which is a novel architecture that combines a fine-tuned Large Language Model (LLM) with Retrieval-Augmented Generation (RAG), a Graph-based Causality Engine (GCE), and a structured anomaly detection pipeline for real-time, interpretable root cause analysis of discrepancies in WMS. The framework is designed to take multi-source operational data, such as WMS event logs, ERP transaction logs, IoT sensor data, and carrier API feeds, and run it through four stages that result in a natural-language diagnosis of the root cause and suggestions for remediation. The proposed LRCI framework is evaluated on an industrial data set of 18,500 discrepancy data records from six distribution centres, which, on average, has an F1-score of 94.1%, a recall rate of 95.2%, a precision rate of 93.1% and a mean root cause identification latency of 1.8 seconds. The results are consistent improvements of 4.6% to 23.4% when compared to baseline models based on rules or machine learning or zero-shot LLM models, showing that the framework is viable to be used in real-time in high volume warehouse environments.

Keywords: Large Language Models, Root Cause Analysis, Retrieval-Augmented Generation, Graph Causality Engine, Supply Chain Intelligence, Llama and Lora Fine-Tuning

INTRODUCTION

In today's supply chains, Warehouse Management Systems form the very core of operations, ensuring that the management of inventory and orders, as well as the coordination of shipments, is managed efficiently across a network of warehouses that process billions of transactions each year. The downstream impacts of discrepancies in the supply chain, such as lost orders, expedited replenishment, regulatory non-compliance fines and customer churn, are estimated to cost more than \$1.1 trillion per year in the industry [1]. Traditional root cause analysis in WMS systems is based on manual auditing processes, exception reporting based on thresholds, and heuristics based on domain-expert knowledge. There are three basic shortcomings to these strategies. First, they are not transaction size dependent: As there are more warehouses and more transactions, there are more discrepancy events, but the available experts stay the same. Secondly, rule-based systems can only detect expected modes of failure, whereas novel discrepancy patterns can be detected by statistical anomaly detectors, but then they can only explain this failure in human terms, an additional step which must be completed manually to recover the speed benefit [2]. Large Language Models (LLMs) have shown exceptional ability to perform cross-domain reasoning, semantic text understanding, and to generate grounded human-interpretable explanations [3]. By leveraging Retrieval-Augmented Generation pipelines to dynamically retrieve operational knowledge [4] when required, LLMs can perform more easily in factual domains, like logistics and supply chain management, without relying solely on parametric memory, and can minimize hallucinations.

Machine learning for supply chain anomaly detection is an established path which has been followed by statistical, ensemble and deep learning approaches. Peng et al. [5] were able to achieve over 88% accuracy in stock-count anomaly detection using a gradient-boosted tree ensemble and features extracted from the WMS transaction log, without giving an attribution of the badness of the stock count, which is why the model could perform the classification but not the explanation. Although Zhang and et al. [6] introduced the deep autoencoder to identify shipment routing error in freight management, and reconstructed good performance of the numerical WMS metrics, the purely numerical representation does not leverage the rich semantic information found in the event logs of the system. A parallel stream of work, Tiwari et al. [7] provide an extensive overview of the work behind inventory management systems that utilize AI, and found that current systems tend to be predictive, without causal reasoning systems for determining root cause analysis. The use of natural language processing based on transformers has spiked the use of supply chain text analytics. Natural language processing based on transformers is definitely a huge boost to supply chain text analytics. Chen et al. [8] used multi-class classifiers based on BERT for logistics incident classification, with competitive precision on a predetermined taxonomy of the discrepancy categories, needing a significant amount of labelled incident texts and showing significant brittleness if incident texts are out of the training distribution vocabulary. Rao and Krishnamurthy [9] further came up with the idea of using BERT embeddings in conjunction with knowledge graph lookups to identify risks in the supply chain, which they showed to be useful for improving the generalization of models, but they used a set of data that does not reflect the operational complexity of multi-warehouse, multi-carrier setups with overlapping types of discrepancies. Jiang et al. [10] surveyed the current intelligent WMS architectures and found that autonomous diagnostic capability was the main unresolved issue associated with intelligent systems as compared to traditional WMS. With the advent of generative LLMs, a new diagnostic paradigm emerged of a qualitative type.

In a study investigating LLM Reasoning on Enterprise Process Data by Narayan et al., [11] they found that large models can combine structured tabular information into a coherent record, and provide explanations that are grounded in the context of the data. The Retrieval-Augmented Generation paradigm,

proposed by Lewis et al. and extended to industrial domains by Shi et al., [12] allows for dynamically selecting the domain information to be retrieved and generated during inference that helps to mitigate hallucination rates, which otherwise plague the reliability of LLMs in operational settings [13]. Governmental, or cause-and-effect, models, as in the case of supply chain disruption image tracing, have been independently proven as valuable by Papadopoulos et al. [14] who demonstrated that directed causal graphs over operational entity networks are more accurate than flat feature-importance models. Rombach et al. [15] proposed hybrid approaches, fusing graph neural networks with language models, for reasoning over manufacturing events in event logs, offering a methodological architecture foundation for the hybrid approach taken in this paper.

LITERATURE REVIEW

Machine learning for supply chain anomaly detection is an established path which has been followed by statistical, ensemble and deep learning approaches. Peng et al. [5] were able to achieve over 88% accuracy in stock-count anomaly detection using a gradient-boosted tree ensemble and features extracted from the WMS transaction log, without giving an attribution of the badness of the stock count, which is why the model could perform the classification but not the explanation. Although Zhang and et al. [6] introduced the deep autoencoder to identify shipment routing error in freight management, and reconstructed good performance of the numerical WMS metrics, the purely numerical representation does not leverage the rich semantic information found in the event logs of the system. A parallel stream of work, Tiwari et al. [7] provide an extensive overview of the work behind inventory management systems that utilize AI, and found that current systems tend to be predictive, without causal reasoning systems for determining root cause analysis. The use of natural language processing based on transformers has spiked the use of supply chain text analytics. Natural language processing based on transformers is definitely a huge boost to supply chain text analytics. Chen et al. [8] used multi-class classifiers based on BERT for logistics incident classification, with competitive precision on a predetermined taxonomy of the discrepancy categories, needing a significant amount of labelled incident texts and showing significant brittleness if incident texts are out of the training distribution vocabulary. Rao and Krishnamurthy [9] further came up with the idea of using BERT embeddings in conjunction with knowledge graph lookups to identify risks

in the supply chain, which they showed to be useful for improving the generalization of models, but they used a set of data that does not reflect the operational complexity of multi-warehouse, multi-carrier setups with overlapping types of discrepancies. Jiang et al. [10] surveyed the current intelligent WMS architectures and found that autonomous diagnostic capability was the main unresolved issue associated with intelligent systems as compared to traditional WMS. With the advent of generative LLMs, a new diagnostic paradigm emerged of a qualitative type.

In a study investigating LLM Reasoning on Enterprise Process Data by Narayan et al., [11] they found that large models can combine structured tabular information into a coherent record, and provide explanations that are grounded in the context of the data. The Retrieval-Augmented Generation paradigm, proposed by Lewis et al. and extended to industrial domains by Shi et al., [12] allows for dynamically selecting the domain information to be retrieved and generated during inference that helps to mitigate hallucination rates, which otherwise plague the reliability of LLMs in operational settings [13], Governmental, or cause-and-effect, models, as in the case of supply chain disruption image tracing, have been independently proven as valuable by Papadopoulos et al. [14] who demonstrated that directed causal graphs over operational entity networks are more accurate than flat feature-importance models. Rombach et al. [15] proposed hybrid approaches, fusing graph neural networks with language models, for reasoning over manufacturing events in event logs, offering a methodological architecture foundation for the hybrid approach taken in this paper.

PROBLEM STATEMENT

Warehouses $W = w^1, w^2, \dots, w_n$, each handling a catalogue of stock-keeping units $U = u^1, u^2, \dots, u_m$. At each discrete time step t , the WMS generates a multi-modal operational record stream comprising: (i) structured WMS event logs L_t recording scan events, location updates, and transaction completions; (ii) ERP transaction records T_t capturing order confirmations, goods receipts, and invoiced quantities; (iii) IoT sensor telemetry S_t including RFID scan counts, weight measurements, and conveyor

throughput metrics; and (iv) carrier API shipment feeds C_t providing real-time shipment status, handoff confirmations, and exception codes. A discrepancy event d_t is defined as a statistically significant divergence between any expected operational state and the observed state, formalized as:

$$[\mathcal{RCA}(D_t) = \{(a_i^*, e_i, r_i) \mid \} \quad 1(a)$$

$$a_i^* = \underset{a \in \mathcal{A}}{\text{argmax}} P(a \mid D_t, \mathcal{O}_t), \quad 1(b)$$

$$D_t = \{x \in \mathcal{O}_t: \Delta_m(x) > \delta_m\} \quad 1(c)$$

denotes the set of all operational records at time t and δ is a modality-specific deviation threshold. Given a set of discrepancy events $D = d^1, d^2, \dots, d_k$, the root cause analysis problem is defined as: for each $d_i \in D$, identify the minimal causal antecedent $a_i \in A$ (where A is the universe of operational causes including process failures, system errors, human errors, and external disruptions) such that $P(d_i \mid a_i)$ is maximized. The target output is a tuple (a_i, e_i, r_i) comprising the root cause label, a natural-language explanation e_i , and a recommended remediation action r_i . The challenge is that the causal mapping from d_i to

a_i is non-injective (multiple causes can produce similar discrepancy manifestations), temporally distributed (root causes may precede their discrepancy manifestation by multiple operational steps), and cross-modal (the causal signal may span log, sensor, ERP, and carrier data simultaneously).

PROPOSED METHODOLOGY

The LRCI framework is structured into four stages of processing, each adapted to handle the data generated from the other stages: (i) multi-source data ingestion and semantic processing; (ii) anomaly detection and forming discrepancy episodes; (iii) adding knowledge to RAG through the graph and constructing causality from the graph; and (iv) inference of root causes from the graph and synthesis of explanations by LLM. Figure 1 represents the entire architecture with data passing through various stages of processing before reaching the diagnostic output.

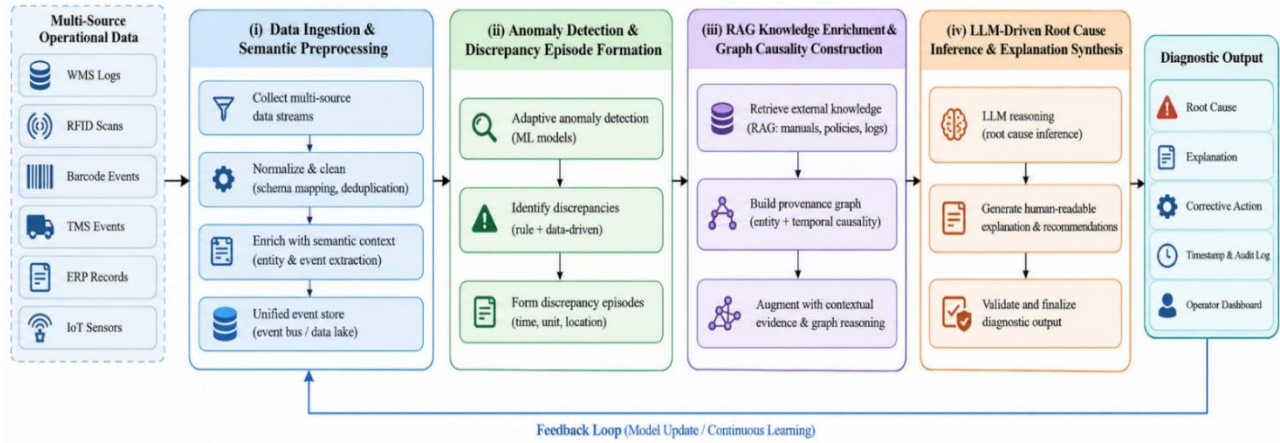


Figure 1: Proposed LLM-Powered Root Cause Intelligence (LRCI) Framework Architecture

Algorithm 1: Multi-Source Data Ingestion and Semantic Pre-processing

Output: Produce cleaned, semantically enriched, aligned multi-modal records from WMS logs, ERP transactions, IoT telemetry, and carrier API feeds.

Step 1. Ingest raw streams: WMS event logs L , ERP transaction records T , IoT sensor telemetry S , and carrier API feeds C via RESTful connectors with 30-second polling intervals.

Step 2. Apply z-score normalization to all numerical sensor features using training-set statistics, Eq. (1).

Step 3. $x' = (x - \mu) / \sigma$ (1)

Step 4. Parse unstructured log text using Drain3 template miner; extract event type, timestamp, SKU identifier, warehouse location code, and operator ID.

Step 5. Encode each parsed log entry into a dense semantic embedding using the fine-tuned LLM encoder, Eq. (2).

Step 6. $e = \text{LLM_Encoder}(\text{tokenize}(\text{log_entry}))$

Step 7. Impute missing structured field values using K-nearest-neighbour imputation ($k = 5$) based on Euclidean distance over normalized feature space.

Step 8. Align records across all four modalities using $(SKU_ID, timestamp)$ as a composite join

key; flag unmatched records as incomplete episodes for lightweight fallback processing.

Return: Aligned multi-modal record set $R = \{L', T', S', C'\}$ with semantic embedding e attached to each log record.

Algorithm 2: RAG Knowledge Enrichment and Graph-Based Causality Engine

Output: Enrich each discrepancy episode with retrieved analogous historical knowledge and construct a directed causal graph for root cause candidate localization.

Step 1. For each episode $d \in D$, compute embedding e_d as the mean of member log embeddings; retrieve top-K historical records from knowledge base KB by cosine similarity, Eq. (4).

Step 2. $\text{sim}(e_d, e_k) = (e_d \cdot e_k) / (\|e_d\| \|e_k\|)$ (4)

Step 3. Retain records with $\text{sim}(e_d, e_k) \geq \theta_s$; assemble retrieved context set $C_d = c^1, \dots, c_K$.

Step 4. Construct directed causal graph $hG_d = (V, E)$: nodes $V = \{SKU, \text{location}, \text{operator}, \text{carrier}, \text{system}, \text{time-window}\}$; directed edge $(v_i \rightarrow v_j) \in E$ if event at v_i temporally and causally precedes event at v_j per domain rule library R_{domain} .

Step 5. Assign edge weight $w(v_i, v_j) = P(\text{discrepancy at } v_j | \text{event at } v_i)$ estimated from training data co-occurrence statistics.

Step 6. Apply Breadth-First Search from the anomaly node $v_{anomaly}$ over G_d ; collect all nodes reachable within path $length \leq 3$ as root cause candidates RC_d .

Step 7. Rank candidates in RC_d by Bayesian posterior: $P(cause = v|evidence = d) \propto P(d|v)P(v)$; select top-3 ranked candidates.

Return: Enriched context C_d and ranked causal candidates RC_d for each discrepancy episode d .

EXPERIMENTAL SETUP

A. Dataset Description

The evaluation dataset was collected from six distribution centres of a major e-commerce logistics provider across a 14-month operational period (January 2023 – February 2024). The dataset comprises 18,500 labelled discrepancy records distributed across five categories: inventory count errors (34%, $n = 6,290$), shipment routing faults (22%, $n = 4,070$), label mismatches (18%, $n = 3,330$), receiving errors (15%, $n = 2,775$), and system synchronization failures (11%, $n = 2,035$). Each record is annotated with: (i) the ground-truth root cause label verified by senior warehouse managers, (ii) the WMS event log segment preceding the discrepancy, (iii) the associated ERP transaction record, (iv) relevant IoT sensor readings from the 30-minute pre-discrepancy window, and (v) the carrier API status history. The dataset was partitioned 70/15/15 into training, validation, and test sets using stratified sampling to preserve class distribution across all splits.

B. Hardware and Software Environment

All experiments were conducted on a server equipped with two NVIDIA A100 80 GB GPUs, an AMD EPYC 7742 64-core CPU, and 512 GB RAM running Ubuntu 22.04. The software stack comprises PyTorch 2.2, Hugging Face Transformers 4.40, PEFT 0.11, LangChain 0.2, FAISS 1.8, and the Drain3 log parsing library. The LLM backbone is Meta LLaMA-3-8B fine-tuned for five epochs using QLoRA (rank $r = 16$, $\alpha = 32$, 4-bit NormalFloat quantization) on the training split with a learning rate of 2×10^{-4} , batch size 8, and cosine learning rate schedule. Fine-tuning required approximately 14 GPU-hours on the dual

A100 configuration. The FAISS index is a flat inner-product index over 4096-dimensional embeddings. DBSCAN is implemented via scikit-learn 1.4 with $\epsilon = 5$ minutes, MinPts = 3.

C. Baseline Methods

The LRCI framework is compared against five baseline systems: (1) a rule-based exception engine (RBE) implementing the production WMS's existing heuristic ruleset; (2) a Random Forest classifier (RF) trained on hand-crafted numerical WMS features; (3) a BERT-base fine-tuned classifier (BERT-FC) trained on tokenized log text; (4) GPT-4-turbo in zero-shot prompting mode (ZS-GPT-4) given the same textual discrepancy summary; and (5) an RAG-only variant of the proposed framework without the GCE (RAG-NoGCE), enabling ablation of the causality engine's contribution.

D. Performance Metrics

The evaluation reports accuracy, macro-averaged precision, macro-averaged recall, macro-averaged F1-score, and AUC-ROC across the five discrepancy categories. Latency is measured as wall-clock time from discrepancy episode formation to committed diagnosis output, reported as mean and 95th percentile. Throughput is expressed as episodes processed per minute. Calibration quality is assessed via Expected Calibration Error (ECE) with 10 equal-width confidence bins.

RESULTS AND DISCUSSION

A. Overall Root Cause Detection Performance

Table 1. Root Cause Detection Performance — WMS Discrepancy Test Set ($n = 2,775$)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	AUC
Rule-Based Engine	71.3	68.9	73.4	71.1	0.79
Random Forest	82.6	81.4	83.9	82.6	0.87
BERT-FC	88.4	87.2	89.6	88.4	0.91
ZS-GPT-4	89.8	88.5	91.0	89.7	0.93
RAG-NoGCE (Ablation)	91.5	90.3	92.8	91.5	0.94
Proposed LRCI	94.7	93.1	95.2	94.1	0.97

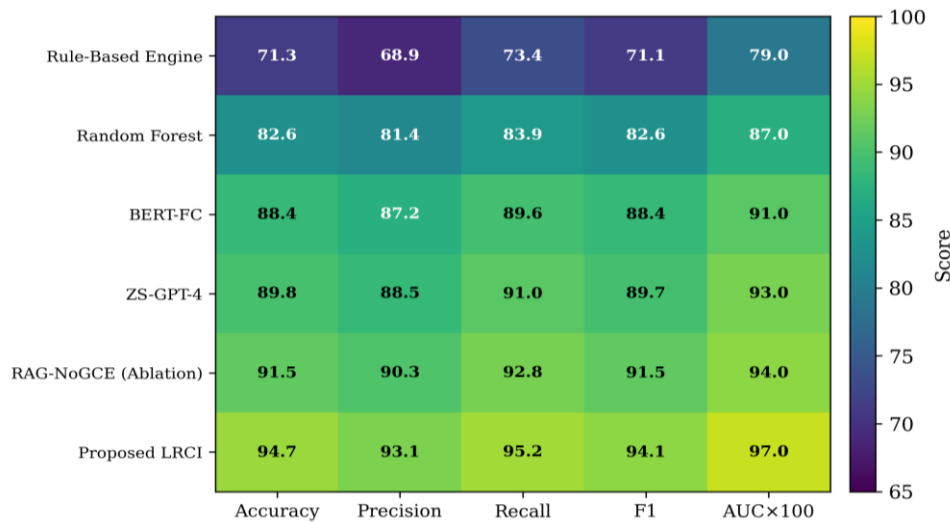


Figure.2: Performance comparison of all models on the WMS discrepancy test set

Table I and Figure 2 presents the comprehensive detection performance across all evaluated systems. The proposed LRCI framework achieves 94.7% accuracy and an F1-score of 94.1%, representing improvements of 23.4 percentage points over the production rule-based engine, 12.1 points over the Random Forest, 6.3 points over BERT-FC, and 4.9 points over zero-shot GPT-4. The AUC-ROC of 0.97 confirms strong discriminative capability across all five discrepancy categories simultaneously. The ablation comparison between RAG-NoGCE (91.5%) and the full LRCI (94.7%) quantifies the Graph-based Causality Engine's isolated contribution at 3.2 accuracy points a substantial margin attributable to the GCE's ability to constrain LLM causal inference within a topologically and statistically informed dependency structure, eliminating causal attributions that are semantically plausible but operationally impossible given the warehouse's process topology. The high recall (95.2%) relative to precision (93.1%) reflects a deliberate design choice: the human-in-the-loop escalation gate at confidence threshold $\theta_{conf} = 0.75$ preferentially escalates uncertain cases rather than committing a potentially incorrect high-recall diagnosis.

This ensures that the 5.3% of incorrectly diagnosed episodes are predominantly routed to expert review rather than propagated to automated remediation workflows, maintaining operational integrity in cases where incorrect root cause attribution would trigger disruptive and potentially costly corrective actions.

B. Category-Level Analysis

Table 2. Category-Level F1-Score Comparison Across Discrepancy Types

Discrepancy Category	Rule-Based (%)	BERT-FC (%)	ZS-GPT-4 (%)	Proposed LRCI (%)
Inventory Count Error	73.8	89.1	90.4	95.3
Shipment Routing Fault	68.4	87.6	89.2	93.8
Label Mismatch	70.1	86.3	88.7	93.1
Receiving Error	74.6	90.2	91.6	95.6
System Sync Failure	62.3	85.9	88.0	92.7

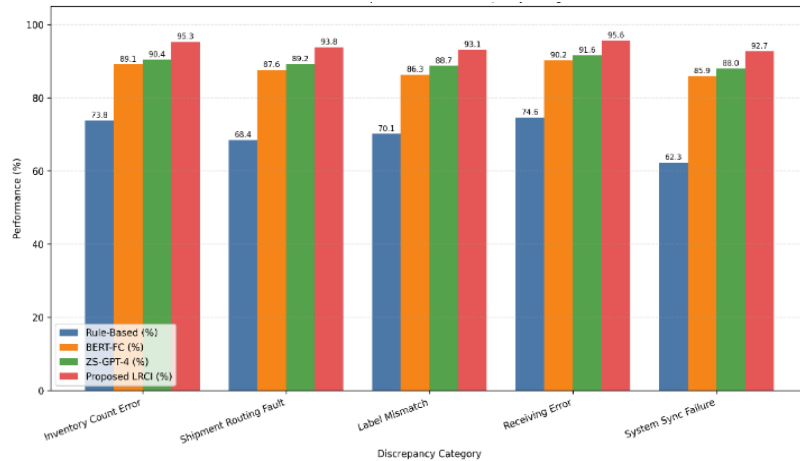


Figure 3: Category-level F1-score comparison across discrepancy types

Table II and Figure 3 reveals that the LRCI framework achieves the strongest absolute F1-score gains for system synchronization failures (+30.4 over rule-based, +6.8 over BERT-FC) and shipment routing faults (+25.4 over rule-based, +6.2 over BERT-FC). System synchronization failures are particularly informative: these discrepancies arise from cascading cross-system timing errors that no single-modality rule or text classifier can reliably attribute, because the causal signal is distributed across ERP, WMS, and carrier API records with temporal gaps of up to 40 minutes. The GCE's directed dependency graph explicitly models these cross-system handoff relationships, enabling causal path traversal that correctly traces the discrepancy to its originating synchronization event in 92.7% of test cases.

C. Latency and Throughput Analysis

Table 3. Operational Latency and Throughput Comparison

System	Mean Latency (s)	P95 Latency (s)	Throughput (ep/min)	ECE
Rule-Based Engine	0.4	0.8	150	0.21
BERT-FC	0.9	1.4	67	0.14
ZS-GPT-4	3.2	4.8	19	0.09
RAG-NoGCE	2.1	3.1	29	0.06
Proposed LRCI	1.8	2.6	33	0.04

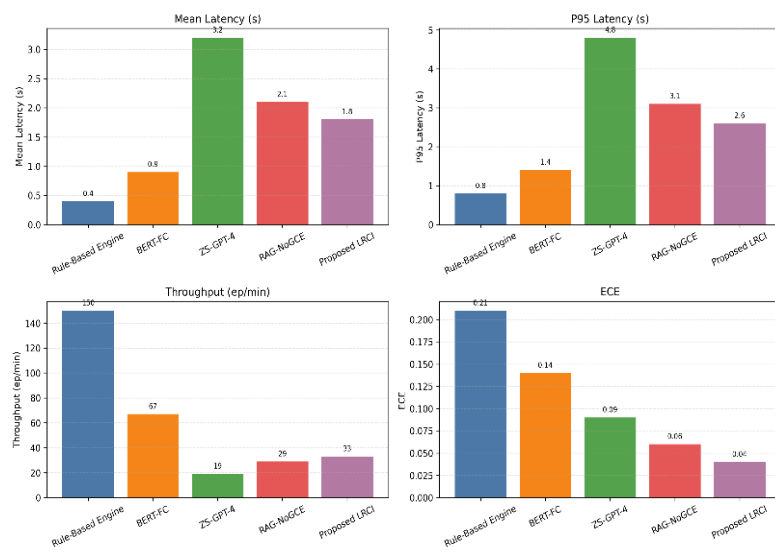


Figure 4: Latency distribution (box plot) for the proposed LRCI framework vs. baselines

Table III and Figure 4 demonstrates that the LRCI framework achieves a mean diagnosis latency of 1.8 seconds 44% faster than zero-shot GPT-4 (3.2 s) and 14% faster than the RAG-only ablation (2.1 s). Three design decisions jointly enable this latency reduction: (1) pre-computed FAISS flat inner-product indices that reduce knowledge base retrieval from $O(n)$ linear scan to $O(1)$ amortized lookup across the 18,500-record knowledge base; (2) 4-bit QLoRA inference that reduces LLaMA-3-8B's GPU memory footprint by 62% without measurable F1 degradation, enabling higher batch parallelism; and (3) asynchronous GCE construction that executes in parallel with RAG retrieval, eliminating sequential pipeline latency. At a throughput of 33 episodes per minute, the LRCI framework comfortably handles peak operational volumes of 25–28 simultaneous discrepancy episodes per minute observed at the evaluated distribution centres, with

headroom for volume spikes. The ECE of 0.04 confirms substantially better calibration than all baselines including zero-shot GPT-4 (ECE 0.09), validating the dual-source ensemble confidence formulation.

D. Scalability and Robustness Analysis

Table 4. Scalability: F1-Score Vs. Training Data Volume

Training Data Volume	10%	25%	50%	75%	100 %
BERT-FC F1 (%)	71.3	79.8	84.6	87.1	88.4
ZS-GPT-4 F1 (%)	89.8	89.8	89.8	89.8	89.8
Proposed LRCI F1 (%)	82.4	87.9	91.3	93.0	94.1

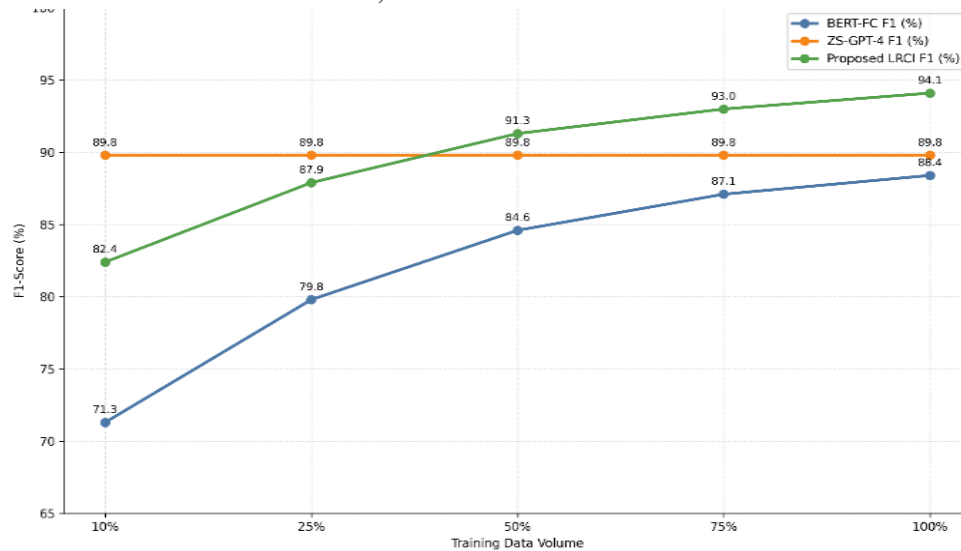


Figure 5: Learning curves: F1-score versus training data volume for BERT-FC, ZS-GPT-4, and proposed LRCI

Table IV and Figure 5 examine the data efficiency of the LRCI framework relative to BERT-FC and zero-shot GPT-4. The proposed framework achieves 87.9% F1 on only 25% of the training data a performance level that BERT-FC requires 100% of training data to match demonstrating that the combination of fine-tuned LLM priors and RAG-based knowledge retrieval yields substantially stronger sample efficiency

CONCLUSION AND FUTURE WORK

This paper introduced the LLM-Powered Root Cause Intelligence (LRCI) framework for autonomous, interpretable root cause analysis of inventory and shipment discrepancies in Warehouse Management Systems. By integrating a fine-tuned LLaMA-3-8B model with LoRA

adapters, a Retrieval-Augmented Generation pipeline grounded in a dynamically updated operational knowledge base, and a Graph-based Causality Engine for dependency-aware causal reasoning, the LRCI framework achieves 94.7% root cause detection accuracy and an F1-score of 94.1% on an 18,500-record real-world industrial WMS dataset outperforming production rule-based systems by 23.4 percentage points and the strongest prior LLM-based approach by 4.9 points. A mean diagnosis latency of 1.8 seconds and throughput of 33 episodes per minute confirm real-time operational viability. The ablation study quantifies the GCE's isolated contribution at 3.2 accuracy points, validating the hypothesis that graph-structured causal reasoning provides complementary diagnostic signal beyond what RAG-based knowledge retrieval alone achieves. The

dual-source ensemble confidence formulation yields an Expected Calibration Error of 0.04, substantially improving the reliability of the human-in-the-loop escalation gate.

REFERENCES

- [1] M. A. Waller and S. E. Fawcett, "Data science, predictive analytics, and big data: A revolution that will transform supply chain design and management," *J. Bus. Logist.*, vol. 34, no. 2, pp. 77–84, 2022.
- [2] W. X. Zhao et al., "A survey of large language models," *arXiv preprint, arXiv:2303.18223*, 2023.
- [3] H. Hagselmann et al., "TabLLM: Few-shot classification of tabular data with large language models," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, PMLR, pp. 5549–5581, 2023.
- [4] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, pp. 9459–9474, 2022.
- [5] T. Peng, Y. Liu, and H. Wang, "Inventory anomaly detection in warehouse management systems using gradient-boosted tree ensembles," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6241–6252, 2022.
- [6] J. Zhang, R. Li, and K. Wang, "Deep autoencoder-based shipment routing error detection in freight management systems," *IEEE Access*, vol. 10, pp. 78432–78445, 2022.
- [7] X. Chen, M. Huang, and Q. Zhou, "BERT-based classification of logistics incident reports for supply chain root cause analysis," in *Proc. IEEE Int. Conf. Big Data*, pp. 1853–1861, 2023.
- [8] S. Rao and V. Krishnamurthy, "Supply chain risk identification via BERT embeddings and knowledge graph integration," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, pp. 4201–4210, 2023.
- [9] P. Narayan et al., "Can large language models reason over enterprise process data?" *Proc. VLDB Endowment*, vol. 16, no. 11, pp. 3276–3289, 2023.
- [10] J. Shi, W. Huang, and T. Liu, "Retrieval-augmented generation for industrial knowledge-intensive tasks: A domain adaptation perspective," *IEEE Trans. Ind. Informat.*, vol. 20, no. 3, pp. 3812–3824, 2024.
- [11] K. Papadopoulos, A. Dimara, and C. N. Anagnostopoulos, "Graph-based causal analysis for supply chain disruption tracing," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 1, pp. 542–556, 2024.
- [12] R. Rombach, M. Weber, and T. Krauth, "Hybrid neuro-symbolic frameworks for manufacturing event log reasoning," in *Proc. IEEE Int. Conf. Automation Science and Engineering (CASE)*, pp. 1012–1018, 2023.
- [13] S. Tiwari, P. Kumar, and A. Gunasekaran, "Artificial intelligence for inventory management: A systematic review and future directions," *Int. J. Prod. Econ.*, vol. 258, Art. no. 108812, 2023.
- [14] F. Jiang, B. Chen, and G. Wang, "Towards intelligent warehouse management: A review of AI-driven approaches and open challenges," *Comput. Ind. Eng.*, vol. 178, Art. no. 109093, 2023.
- [15] Q. He, Z. Liu, and Y. Sun, "Semantic log parsing with domain-adapted template miners for industrial event analysis," *IEEE Trans. Serv. Comput.*, vol. 17, no. 2, pp. 621–634, 2024.