



OPEN ACCESS

Volume: 5

Issue: 4

Month: October

Year: 2026

ISSN: 2583-7117

Published: 01.10.2026

Citation:

Rajesh Kumar Sharma, Priya Venkata-
raman, Suresh Narayanan “LLM-Pow-
ered Root Cause Intelligence for Inven-
tory and Shipment Discrepancies in
Warehouse Management Systems” Inter-
national Journal of Innovations in Sci-
ence 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

Rajesh Kumar Sharma¹, Priya Venkataraman², Suresh Narayanan³

¹Assist Professor, Dept of Computer Science & Engineering Jain University, Bangalore, India

²Associate Professor, Dept of Information Technology Jain University, Bangalore, India

³Assist Professor, Dept of Artificial Intelligence & ML Jain University, Bangalore, India

Abstract

Inventory and shipment discrepancies in Warehouse Management Systems (WMS) impose significant operational costs on global supply chains, yet current diagnostic approaches remain fragmented, reactive, and heavily reliant on domain-expert intervention. This paper proposes the LLM-Powered Root Cause Intelligence (LRCI) framework, a novel architecture that unifies 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 WMS discrepancies. The framework ingests multi-source operational data — WMS event logs, ERP transaction records, IoT sensor telemetry, and carrier API feeds — and processes them through a four-stage pipeline culminating in LLM-generated root cause diagnoses with natural-language explanations and remediation recommendations. Evaluated on an industrial dataset of 18,500 discrepancy records across six distribution centres, the proposed LRCI framework achieves 94.7% accuracy, 93.1% precision, 95.2% recall, and an F1-score of 94.1%, with a mean root cause identification latency of 1.8 seconds. These results represent consistent improvements of 4.6 to 23.4 percentage points over rule-based, machine-learning, and zero-shot LLM baselines, demonstrating the framework's practical viability for real-time deployment in high-volume warehouse environments.

Keywords; Large Language Models, Root Cause Analysis, Inventory Discrepancies, Shipment Discrepancies, Warehouse Management Systems, Retrieval-Augmented Generation, Graph Causality Engine, Supply Chain Intelligence, LLaMA, LoRA Fine-Tuning

INTRODUCTION

Warehouse Management Systems constitute the operational nucleus of contemporary supply chains, orchestrating inventory tracking, order fulfilment, and shipment coordination across distributed facilities that handle billions of transactions annually. Despite pervasive automation, a persistent class of failures — inventory count mismatches, mislabeled shipments, receiving errors, and system synchronization failures — continues to erode operational efficiency at scale. Industry estimates place the cumulative cost of supply chain discrepancies at over \$1.1 trillion annually, driven by downstream ripple effects including order cancellations, expedited replenishment, regulatory non-compliance penalties, and customer churn [1]. The identification of the precise root cause behind each discrepancy — rather than merely flagging its existence — is the diagnostic capability that converts anomaly detection from an alerting function into a decision-support capability.

Conventional root cause analysis in WMS environments relies on manual auditing workflows, threshold-based exception reporting, and domain-expert heuristics. These approaches suffer from three fundamental limitations. First, they do not scale with transaction volume: as warehouse throughput grows, the number of discrepancy events grows proportionally while expert availability remains fixed. Second, rule-based systems capture only anticipated failure modes; novel discrepancy patterns — introduced by carrier changes, system upgrades, or seasonal demand anomalies — evade predefined rules entirely. Third, statistical anomaly

detectors identify that a deviation has occurred but cannot articulate why it occurred in terms intelligible to warehouse operators, necessitating further manual investigation that negates much of the speed advantage [2].

Large Language Models have demonstrated a remarkable capacity for cross-domain reasoning, semantic text understanding, and the generation of grounded, human-interpretable explanations [3]. When augmented with Retrieval-Augmented Generation pipelines that supply dynamically retrieved operational knowledge [4], LLMs can ground their inferences in domain-specific facts rather than relying solely on parametric memory, substantially reducing hallucination in factual domains such as logistics and supply chain management. The integration of LLMs with graph-structured causal models further extends this capability to multi-hop relational reasoning — precisely the kind of inference required to trace a shipment discrepancy through a chain of warehouse process steps, system handoffs, and carrier interactions to its originating cause.

Motivated by these convergent advances, this paper introduces the LLM-Powered Root Cause Intelligence (LRCI) framework, which systematically integrates fine-tuned LLM inference, RAG-based knowledge enrichment, and a Graph-based Causality Engine to deliver autonomous, real-time root cause diagnoses for inventory and shipment discrepancies in WMS environments. The remainder of this paper is organized as follows. Section II reviews the relevant literature.

LITERATURE REVIEW

The application of machine learning to supply chain anomaly detection has a well-documented trajectory spanning statistical, ensemble, and deep learning paradigms. Peng et al. demonstrated that gradient-boosted tree ensembles applied to WMS transaction log features can identify stock-count anomalies with accuracy above 88%, but produced no causal attribution — the model classifies but does not explain [5]. Zhang and colleagues proposed a deep autoencoder for detecting shipment routing errors in freight management, achieving strong reconstruction-based anomaly scores on numerical WMS metrics, but the entirely numerical representation fails to exploit the rich semantic content of operational event logs [6]. In a parallel stream of work, Tiwari et al. conducted a comprehensive review of AI-driven inventory management approaches, concluding that existing systems are predominantly predictive rather than diagnostic, lacking the causal reasoning infrastructure necessary for root cause identification [13].

The adoption of transformer-based natural language processing elevated supply chain text analytics considerably. Chen et al. applied BERT-based multi-class classifiers to logistics incident reports, achieving competitive precision on a fixed taxonomy of discrepancy categories, but requiring substantial labeled training data and exhibiting pronounced brittleness when incident descriptions deviate from training distribution vocabulary [7]. Rao and Krishnamurthy extended this direction by coupling BERT embeddings with knowledge graph lookups for supply chain risk identification, demonstrating that external structured knowledge meaningfully improves model generalization, though their evaluation dataset does not capture the operational complexity of multi-warehouse, multi-carrier environments with concurrent discrepancy modes [8]. Jiang et al. reviewed the landscape of intelligent WMS architectures and identified autonomous diagnostic capability as the principal unsolved challenge differentiating emerging intelligent systems from legacy WMS platforms [14].

The emergence of generative LLMs introduced a qualitatively different diagnostic paradigm. Investigations by Narayan et al. into LLM reasoning over enterprise process data demonstrated that large models can synthesize structured tabular records and generate contextually grounded explanations, provided appropriate prompt engineering is applied [9]. The Retrieval-Augmented Generation paradigm, consolidated by Lewis et al. and subsequently adapted for domain-specific industrial contexts by Shi et al., enables LLMs to retrieve and incorporate dynamically selected domain knowledge at inference time, substantially reducing the hallucination rate that otherwise limits LLM reliability in operational settings [15], [10]. Graph-based causality modeling independently established its value for supply chain disruption tracing through Papadopoulos et al., who showed that directed causal graphs over operational entity networks yield more precise disruption attribution than flat feature-importance methods [11]. Hybrid neuro-symbolic frameworks combining graph neural networks and language models for manufacturing event log reasoning were explored by Rombach et al., providing a methodological precedent for the architectural fusion pursued in this paper [12].

PROBLEM STATEMENT

Let a Warehouse Management System operate over a set of 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:

$$d_t = r \in R_t: |observed(r) - expected(r)| > \delta, \text{ where } R_t \quad (1)$$

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 $P(d_i|a_i)$ 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).

RESEARCH GAP AND RESEARCH CONTRIBUTIONS

Research Gap

A synthesis of the reviewed literature reveals three specific gaps that collectively motivate this work. First, no existing WMS diagnostic system integrates an LLM reasoning backbone with structured operational data from multiple simultaneous sources (logs, ERP, IoT, carrier). Current systems either apply NLP to textual incident reports in isolation or apply statistical models to structured sensor data in isolation, forgoing the cross-modal synthesis that characterizes the most informative discrepancy patterns. Second, the RAG paradigm has not been applied to WMS root cause analysis: current diagnostic systems rely on fixed rule libraries or static training corpora that cannot incorporate newly discovered discrepancy patterns without model retraining. Third, graph-structured causality

reasoning which explicitly models the directed dependency relationships among operational entities — has not been coupled with LLM inference for WMS diagnosis. Without a causal graph, LLMs must infer causal relationships from linguistic context alone, a task for which they are poorly calibrated in the absence of domain-specific grounding.

Research Contributions

The principal contributions of this paper are as follows:

(1) LLM-Powered Root Cause Intelligence (LRCI) Framework: A novel end-to-end architecture that integrates fine-tuned LLM inference, RAG-based knowledge enrichment, and a Graph-based Causality Engine for autonomous, interpretable WMS discrepancy root cause analysis — the first such integration in the warehouse management literature.

(2) WMS-Specific LLM Fine-Tuning Protocol: A parameter-efficient LoRA-based fine-tuning protocol for LLaMA-3-8B on a curated WMS operational corpus, producing a domain-adapted model that achieves 94.7% diagnostic accuracy on the evaluation dataset.

(3) Graph-based Causality Engine (GCE): A directed operational dependency graph construction and traversal algorithm that identifies root cause candidates through Bayesian posterior scoring over causal paths, contributing 3.2 accuracy points over RAG-only inference.

(4) Adaptive Anomaly Detection Pipeline: A Mahalanobis distance-based multi-modal anomaly detector with a rolling adaptive threshold that maintains calibrated sensitivity across seasonal and volumetric operational shifts.

(5) Empirical Validation: A rigorous evaluation on an 18,500-record real-world industrial WMS dataset demonstrating state-of-the-art performance with a 1.8-second mean diagnosis latency, confirming operational viability.

PROPOSED METHODOLOGY

The LRCI framework is organized into four coordinated processing stages: (i) multi-source data ingestion and semantic preprocessing; (ii) adaptive anomaly detection and discrepancy episode formation; (iii) RAG knowledge enrichment and graph causality construction; and (iv) LLM-driven root cause inference and explanation synthesis. Figure 1 illustrates the complete framework architecture, showing data flows from raw operational streams through processing stages to the final diagnostic output.

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

Algorithm 1: Multi-Source Data Ingestion and Semantic Preprocessing

Algorithm 1: Multi-Source Data Ingestion and Semantic Preprocessing	
Output: Produce cleaned, semantically enriched, aligned multi-modal records from WMS logs, ERP transactions, IoT telemetry, and carrier API feeds.	
Ste p 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.
Ste p 2.	Apply z-score normalization to all numerical sensor features using training-set statistics, Eq. (1).
Ste p 3.	$x' = (x - \mu) / \sigma$ (1)
Ste p 4.	Parse unstructured log text using Drain3 template miner; extract event type, timestamp, SKU identifier, warehouse location code, and operator ID.
Ste p 5.	Encode each parsed log entry into a dense semantic embedding using the fine-tuned LLM encoder, Eq. (2).
Ste p 6.	$e = \text{LLM_Encoder}(\text{tokenize}(\text{log_entry}))$
Ste p 7.	Impute missing structured field values using K-nearest-neighbour imputation (k = 5) based on Euclidean distance over normalized feature space.
Ste p 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.
Ret urn:	Aligned multi-modal record set $R = \{L', T', S', C'\}$ with semantic embedding e attached to each log record.

Eq. (1) standardizes each numerical sensor measurement using the training-set mean μ and standard deviation σ , ensuring that features measured on heterogeneous scales RFID scan counts, pallet weights in kilograms, conveyor speeds in units per hour contribute proportionally to downstream distance computations without scale-induced bias.

This normalization step is modality-specific: separate (μ , σ) pairs are maintained for each sensor type and warehouse zone, capturing the distinct operational distributions across facility regions.

Eq. (2) encodes each parsed log entry through the fine-tuned LLM encoder, producing a 4096-dimensional semantic embedding e that captures operational context beyond surface keyword matching. These embeddings index the FAISS knowledge base and drive the cosine similarity retrieval in Algorithm 3, enabling semantic rather than lexical matching of analogous historical incidents.

Algorithm 2: Adaptive Anomaly Detection and Discrepancy Episode Formation

Algorithm 2: Adaptive Anomaly Detection and Discrepancy Episode Formation	
Output: Detect anomalous records, classify them into discrepancy categories, and cluster temporally proximate anomalies into coherent discrepancy episodes.	
Ste p 1.	For each record $r \in R$, compute the Mahalanobis anomaly score $A(r)$ against the modality-specific reference distribution parameterized by (μ_{ref}, Σ) , Eq. (3).
Ste p 2.	$A(r) = \sqrt{(r - \mu_{ref})^T \Sigma^{-1} (r - \mu_{ref})}$ (3)
Ste p 3.	Flag record r as anomalous if $A(r) > \theta_A$, where adaptive threshold θ_A is the rolling 97.5th percentile of $A(r)$ over the preceding 7-day window; recompute θ_A daily.
Ste p 4.	Apply fine-tuned BERT-based discrepancy classifier (5-class) to flagged records; assign preliminary category label from $\text{inventory_count_error}, \text{shipment_routing_fault}, \text{label}_m$
Ste p 5.	Cluster temporally proximate anomalous records using DBSCAN ($\epsilon = 5min, MinPts = 3$) to form discrepancy episodes; assign episode ID and aggregate member records.
Ste p 6.	For each episode, compute episode severity score: $sev(d) = \max_{r \in d} A(r) \times d $, where $ d $ is the episode cardinality; prioritize high-severity episodes for immediate LLM processing.
Ret urn:	Discrepancy episode set $D = \{d^1, d^2, \dots, d_n\}$ each annotated with anomaly scores, preliminary category labels, severity score, and member records.

Eq. (3) computes the Mahalanobis distance of each operational record from the modality-specific reference distribution, incorporating the full feature covariance matrix Σ . Unlike Euclidean-distance anomaly scores that treat all features as independent and equally scaled, the Mahalanobis formulation accounts for correlated WMS metrics: for instance, a simultaneous drop in RFID scan count and weight sensor reading is a far stronger discrepancy signal than either deviation alone, and the covariance matrix encodes this dependency. The adaptive threshold θ_A recomputed as the rolling 97.5th percentile over the preceding seven-day period — ensures that the detector remains calibrated against seasonal volume shifts, promotional surges, and facility configuration changes without requiring manual threshold adjustment.

Algorithm 3: RAG Knowledge Enrichment and Graph-Based Causality Engine

Algorithm 3: 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.	$sim(e_d, e_k) = (e_d \cdot e_k) / (\ e_d\ \ e_k\)$ (4)
Step 3.	Retain records with $sim(e_d, e_k) \geq \theta_{sim} = 0.82$; assemble retrieved context set $C_d = c^1, \dots, c_K$.
Step 4.	Construct directed causal graph $hG_d = (V, E)$: nodes $V = \{SKU, \text{location, operator, carrier, system, time-window}\}$; 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 ≤ 3 as root cause candidates RC_d .
Step 7.	Rank candidates in RC_d by Bayesian posterior: $P(\text{cause} = v \text{evidence} = d) \propto P(d v)P(v)$; select top-3 ranked candidates.
Return:	Enriched context C_d and ranked causal candidates RC_d for each discrepancy episode d .

Eq. (4) computes cosine similarity between the semantic embedding of the current discrepancy episode and each candidate record in the knowledge base, enabling retrieval of operationally analogous historical incidents regardless of surface-level textual differences. The threshold $\theta_{sim} = 0.82$ was determined through grid search on the validation set, balancing retrieval precision (lower false positives at higher thresholds) against recall (more comprehensive context at lower thresholds). The knowledge base is continuously updated with confirmed root cause diagnoses every high-confidence LLM output is appended to KB after operator acceptance, creating a self-reinforcing diagnostic memory that improves retrieval quality over time.

The Graph-based Causality Engine (GCE) constructs a directed operational dependency graph whose topology encodes both domain expert knowledge (encoded in R_{domain}) and data-driven co-occurrence patterns. The domain rule library R_{domain} captures warehouse process constraints for example, that a receiving scan must precede a putaway event for the same SKU, and that a missing receiving scan causally explains a subsequent inventory count discrepancy. Bayesian posterior scoring over the candidate set ranks causes by the product of their likelihood given the observed evidence and their prior probability estimated from historical incident rates, producing a statistically principled root cause ranking rather than a purely heuristic one.

Algorithm 4: LLM-Driven Root Cause Inference and Explanation Generation

Algorithm 4: LLM-Driven Root Cause Inference and Explanation Generation	
Output: Generate a natural-language root cause diagnosis, confidence score, evidence chain, and remediation recommendation for each discrepancy episode.	
Step 1.	Construct chain-of-thought prompt P_d : $P_d = [\text{System_role}] + [\text{Discrepancy_summary}(d)] + [\text{Anomaly_metrics}(d)] + [\text{Retrieved_context}(C_d)] + [\text{Causal_candidates}(RC_d)] + [\text{Reasoning_instruction}]$.
Step 2.	Submit P_d to fine-tuned LLaMA-3-8B (LoRA adapters, rank $r = 16$, $\alpha = 32$); decode response with temperature $\tau = 0.2$ and top-p = 0.9 for near-deterministic output.
Step 3.	Parse structured fields from LLM response: $\text{root_cause_label, confidence_score, evidence_chain, recommendation}$.
Step 4.	Compute ensemble confidence: $\text{conf_ensemble} = 0.6 \times \text{conf_LLM} + 0.4 \times P_Bayes(\text{top candidate from } RC_d)$.
Step 5.	If $\text{conf_ensemble} < \theta_{conf} = 0.75$, route episode to human-in-the-loop review queue; else commit diagnosis to WMS audit log and trigger remediation workflow.
Step 6.	On operator confirmation of human-reviewed diagnoses, append confirmed record to knowledge base KB; update GCE edge weights via incremental Bayesian update.
Return:	Diagnosis tuple (root_cause_label, confidence_score, evidence_chain, recommended_action) per discrepancy episode.

The chain-of-thought prompt structure in Step 1 adopts a four-stage reasoning template that instructs the LLM to (a) restate the observed anomaly in operational terms, (b) evaluate each causal candidate from RC_d against the retrieved historical context C_d , (c) select the most evidentially supported root cause and justify the selection, and (d) propose a specific, actionable remediation step drawn from the warehouse's standard operating procedure taxonomy.

This structured instruction contrasts with unconstrained free-form generation, which produces more verbose but less operationally precise outputs in WMS contexts.

The ensemble confidence formulation in Step 4 combines the LLM's self-reported confidence score calibrated on the validation set using temperature scaling with the Bayesian posterior probability from the GCE, down-weighting cases where the two sources of evidence disagree. This dual-source confidence estimate is substantially better calibrated than LLM confidence alone (Expected Calibration Error reduced from 0.11 to 0.04 on the validation set), providing a more reliable trigger for the human-in-the-loop escalation gate.

EXPERIMENTAL SETUP

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 labeled 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.

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.

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.

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

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 1 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 $\theta_{\text{conf}} = 0.75$ 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.

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 2 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.

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 3 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.

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 4 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. This characteristic is operationally significant for deployment in new warehouse facilities where labeled discrepancy data may be limited during initial operation. The zero-shot GPT-4 curve is flat by design (no

training data utilized), confirming that even a powerful general-purpose LLM without WMS-specific adaptation cannot match a domain-fine-tuned model at full data availability.

Comparison with Prior Work

Table 5: Comparison with Prior Work on WMS Discrepancy Diagnosis

Study	Year	Dataset	Best Accuracy (%)	Approach
Peng et al. [5]	2022	WMS Transaction Logs	88.3	Gradient Boosted Trees
Zhang et al. [6]	2022	Freight Routing Logs	85.7 (AUC 0.89)	Deep Autoencoder
Chen et al. [7]	2023	Logistics Incident Reports	88.4	BERT-FC
Rao & Krishnamurthy [8]	2023	SC Risk Dataset	90.1	BERT + Knowledge Graph
Proposed LRCI	2024	Industrial WMS (6 DCs, 18.5K)	94.7	LLM + RAG + GCE

Table 5 situates the LRCI framework against the closest prior studies. The proposed method achieves a 4.6-percentage-point accuracy improvement over the strongest prior work (Rao and Krishnamurthy, 90.1%), while operating on a substantially larger and more operationally realistic dataset (18,500 records across six distribution centres versus single-site or synthetic datasets used in prior studies). The accuracy advantage is attributable to three innovations absent from prior work: (1) a fine-tuned generative LLM backbone with warehouse-specific LoRA adaptation; (2) a dynamic RAG pipeline that continuously incorporates confirmed diagnoses into the retrieval knowledge base; and (3) a GCE that imposes warehouse process topology constraints on causal inference, eliminating process-impossible root cause attributions.

CONCLUSION AND FUTURE WORK

Conclusion

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.

Future Work

Several directions merit investigation in subsequent work. First, extending the GCE to dynamic graph structures that adapt in real time to warehouse layout reconfiguration — such as zone reassignments or new conveyor routes introduced by facility expansion — would reduce the frequency of domain rule library updates that currently require manual intervention. Second, the RAG pipeline currently retrieves from a single-warehouse knowledge base; extending it to a federated multi-warehouse retrieval architecture would allow discrepancy patterns observed at one facility to inform diagnoses at others, accelerating knowledge accumulation for newly commissioned warehouses. Third, this paper addresses post-hoc root cause attribution; adapting the framework for proactive discrepancy prediction — identifying process conditions that are likely to generate discrepancies before they manifest — represents a significant extension with substantial operational value. Fourth, multi-lingual log parsing and LLM inference would broaden applicability to international distribution networks where WMS documentation and operator logs are generated in languages other than English. Fifth, a formal energy efficiency analysis of the 4-bit QLoRA inference path versus full-precision alternatives would quantify the sustainability implications of deploying large-scale LLM inference in warehouse environments.

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. Hegselmann 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.