Shifu: A Self-Learning Framework for Automating Root Cause Analysis in Logistics Operations

Yizhe Xu[1], David Maoujoud, Marco Bocchetti, Ruochen Li, Ryan Valoris, Alexandru Amihalachioaei, Menna Elkashef, Xiaoyu Mao
Amazon
Seattle, Washington, USA

ABSTRACT

Modern logistics networks face a critical challenge in performance documentation that consumes substantial resources yet suffers from inconsistent quality, limited expert review, and context-specificity. We present Shifu, an adaptive knowledge acquisition system for automated root cause analysis that learns continuously from operational feedback without requiring gold standard examples. Shifu integrates targeted machine learning, agent-based data analysis, utility-driven insight prioritization, and active learning through a comprehensive feedback loop. We evaluated Shifu across five North American logistics facilities over a two-week deployment, demonstrating improvements in content quality (reaching 87.9% acceptance within one week), effective feedback incorporation (89.5% closure rate), and knowledge expansion (44% metric growth in key categories). Our results show a 4X improvement over baseline systems, with Shifu self-adapting to facility-specific operational contexts while continuously enhancing its analytical capabilities. This approach transforms resource-intensive analytical processes by complementing rather than replacing human expertise, providing a blueprint for continuous learning systems in domains with subjective quality criteria, specialized operational contexts, and limited supervision.

CCS CONCEPTS

• **Computing methodologies → Active learning settings**; **Learning from critiques**; Causal reasoning and diagnostics; • **Information systems → Data analytics**; • **Applied computing → Transportation**; • **Human-centered computing** → Reputation systems.

KEYWORDS

Root Cause Analysis, Knowledge Acquisition, Logistics Optimization, Active Learning, Continuous Feedback Systems, Adaptive Documentation, Human-AI Collaboration, Operational Analytics, Facility-Specific Context, Performance Documentation

ACM Reference Format:
Yizhe Xu[1], David Maoujoud, Marco Bocchetti, Ruochen Li, Ryan Valoris, Alexandru Amihalachioaei, Menna Elkashef, Xiaoyu Mao, . 2025. Shifu: A Self-Learning Framework for Automating Root Cause Analysis in Logistics Operations. In *Proceedings of the 1st Workshop on "AI for Supply Chain: Today and Future" @ 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/XXXXXX.XXXXXX

1 INTRODUCTION

Modern logistics networks require systematic documentation of performance deviations and corrective actions—a valuable cognitive tool [16] that increasingly overwhelms operators as data volume doubles every two years [11]. Traditional supervised learning approaches fail to automate this process due to four critical challenges:

- **Documentation quality** varies significantly across organizations (nearly half compared to best performers) due to differences in skills, tools, and methodologies [12].
- Only a small fraction of reports (less than one-fifth) receive adequate expert review [22], with minimal awareness of knowledge transfer status [12].
- **No "golden standard"** exists for training AI systems, with persistent process mismatches and differing vocabularies across organizations [12].
- **Context specificity** creates transfer challenges in majority of multi-site operations [10], requiring facility-specific documentation despite similar performance metrics.

While measurement is fundamental to logistics improvement [6], the absence of standardized approaches significantly complicates AI-based automation efforts.

To address these logistics documentation challenges, we developed Shifu with three key contributions:

- An **adaptive knowledge acquisition system** that functions without gold standards, doubling its metric understanding within 11 days of deployment—compared to two months of expert involvement in previous systems—through a self-reinforcing operational learning cycle.
- A **structured methodology for root cause analysis** that preserves cognitive value by maintaining analytical exercises for experts, adapting to facility-specific contexts, and incorporating active feedback mechanisms—eliminating thousands of documentation hours while preserving analytical rigor.
- An **evaluation framework for domains with subjective quality criteria** that tracks content evolution at the sentence level, conducts parallel assessment of operational utility and analytical quality, and quantifies learning capacity through continuous use.

KDD '25, August 3, 2025, Toronto, ON, Canada.
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1454-2/25/08.
https://doi.org/10.1145/XXXXXX.XXXXXX

Shifu demonstrates how AI can transform resource-intensive analytical processes by complementing rather than replacing human expertise—reducing operational burden while preserving valuable cognitive processes in logistics operations.

2 RELATED WORK

Expert-Driven vs. Data-Driven Knowledge Acquisition. Root cause analysis systems typically follow either expert-driven approaches requiring significant knowledge codification [1, 9, 14] or data-driven methods extracting historical patterns [2, 19]. The former offer interpretability but lack adaptability; the latter generate insights operators struggle to trust—creating a persistent implementation gap in operational environments.

Static vs. Adaptive Knowledge. Conventional solutions depend on predefined knowledge bases. Systems like Annotate-LLM [3] and LogRCA [20] require either labeled data or predefined causes—approaches that fail where features emerge gradually, labeled data is scarce, and root causes continuously evolve. With only one analysis examined per incident, traditional models cannot learn from these sparse validation signals.

Causality vs. Correlation. RCA methods divide between correlation approaches (efficient but vulnerable to spurious relationships) and causal techniques (rigorous but data-intensive). Current correlation methods require unavailable ground truth validation [5] or human interpretation [4]. While causal methods using directed acyclic graphs [15] or Bayesian Networks offer reliability, they are vulnerable to incomplete input variables—an understated limitation. Frameworks like PyRCA and ProRCA [7, 13] attempt causal knowledge generation but impose strict data requirements most operations cannot meet.

Shifu addresses these limitations through a knowledge system that continuously refines through feedback, augments human decision-making, and systematically collects data enabling future causal approaches.

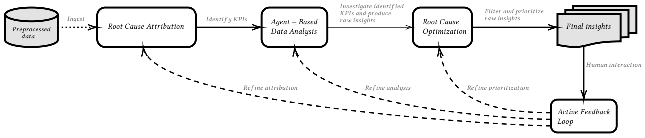

Figure 1: Shifu Architecture

3 SHIFU REASONING ARCHITECTURE

Shifu integrates data acquisition, reasoning, and presentation into a system for automated operational analytics in logistics environments. While data pipelines and interpretable presentation layers are essential components, it is the reasoning architecture that constitutes Shifu's primary contribution to knowledge discovery in logistics performance analysis. We designed this architecture to address the documentation challenges identified in Section 1 through systematic, adaptive, and context-aware reasoning.

The system's data infrastructure implements a bifurcated collection strategy: aggregated Key Performance Indicators (KPIs) such as on-time delivery percentage, transit time, and service time are preserved in a 92-day historical window, while granular transaction-level data is retrieved selectively to optimize computational resources. Our preprocessing pipeline standardizes heterogeneous inputs while preserving data lineage—critical for reproducibility in environments where inconsistent dimension encoding can compromise analysis integrity.

Our reasoning architecture comprises four specialized components working in concert to continuously improve operational insights without manual reprogramming:

1. **Root Cause Attribution**: Identifies key performance drivers through targeted machine learning and explainability techniques.
2. **Agent-Based Data Analysis**: Investigates identified drivers through automated exploration of detailed operational data.
3. **Root Cause Optimization**: Filters and prioritizes insights based on learned utility criteria, actionability, and impact.
4. **Active Feedback Loop**: Continuously refines system behavior based on operator interactions.

3.1 Root Cause Attribution

Motivation: Operations teams need to rapidly identify which metrics most significantly influence target performance indicators, but face challenges with context-specific importance and varying baseline expectations across facilities. We designed the attribution component to provide station-specific analysis that aligns with operators' mental models of their unique operational contexts.

System Components: The attribution module consists of: (1) a feature preprocessing pipeline that normalizes operational metrics, (2) a station-specific supervised model framework, and (3) an explainability layer that quantifies feature contributions.

Technical Implementation: For each station $s \in \mathcal{S}$ and target metric $m \in \mathcal{M}$, we define a feature vector $X_{s,t}$ of operational KPIs and target value $y_{s,t,m}$ at time t, using a 92-day historical window. We construct a supervised model f_s such that $\hat{y}_{s,t,m} = f_s(X_{s,t})$, implemented via AutoGluon TabularPredictor [8] with the `medium_quality` preset to satisfy strict temporal constraints. For models achieving R^2 better than established threshold, we apply SHAP analysis using each station's historical "good performance days" as the baseline rather than global averages, generating station-specific feature importance values that better align with local operational context.

3.2 Agent-Based Data Analysis

Motivation: Once key performance drivers are identified, operators need detailed understanding of the underlying patterns in transaction-level data. Manual deep-dive analysis is time-consuming and inconsistent, requiring automation that can adapt to different data types while maintaining analytical rigor.

System Components: Our multi-agent analysis framework consists of: (1) a data analysis agent that executes parametrized exploratory workflows, (2) a validation agent that independently verifies metric calculations, and (3) an insight generation module that transforms statistical findings into natural language explanations.

Technical Implementation: The data analysis agent applies type-specific analysis protocols (frequency distribution for categorical variables, distribution analysis for numerical variables, and cluster analysis for geo-spatial data). Analysis results are standardized through statistical operations, while the validation agent independently reproduces metric calculations using raw transaction data to ensure integrity. This approach both verifies data quality and provides natural language explanations of complex metrics for operational context.

3.3 Root Cause Optimization

Motivation: Raw analytical insights often overwhelm operators with information that varies in relevance, actionability, and impact. We address this challenge through a learning system that filters and prioritizes insights based on operational value criteria extracted from actual usage patterns.

System Components: The optimization architecture includes: (1) an importance judgment module that filters insights based on learned utility criteria, (2) a priority ranking system that orders insights along multiple dimensions, and (3) a contextual adaptation layer that adjusts output based on operational feedback.

Technical Implementation: The importance judgment employs contrastive Chain-of-Thought reasoning with dual example banks (positive examples from promoted/added insights and negative examples from demoted/deleted insights). These examples undergo vector embedding and K-means clustering to select diverse representatives. The priority ranking implements an LLM-as-Judge architecture trained on pairwise comparisons from operator interactions, evaluating insights along actionability, evidential support, and impact scale dimensions:

Algorithm 1 Insight Prioritization

```
Input: New insight $I$, Example banks $P$ (positive) and $N$ (negative)
function InsightPrediction($I$, $P$, $N$)
    $dimensions$ ← ["actionability", "evidence", "impact"]
    $P_{samples}$ ← SampleDiverse($P$, $k = 3$)
    $N_{samples}$ ← SampleDiverse($N$, $k = 3$)
    $utility$ ← ContrastiveJudgement($I$, $P_{samples}$, $N_{samples}$)
    if $utility$ > threshold then
        $scores$ ← EvaluateAlongDimensions($I$, $dimensions$)
        return $I$ with priority $scores$
    else
        Filter out $I$
    end if
end function
```

3.4 Active Feedback Loop

Motivation: Static analysis systems quickly become irrelevant in dynamic operational environments. Our active feedback loop transforms daily operational interactions into continuous system improvements without requiring dedicated expert support sessions.

System Components: The feedback system consists of: (1) interaction capture mechanisms for four signal types, (2) a dual-channel feedback processor that balances operational and expert input, and (3) a signal routing system that directs feedback to appropriate system components.

Technical Implementation: The system captures structured events containing action type (promote or demote/delete/edit/add), affected content, timestamp, and contextual metadata including user role. These events undergo validation filtering to eliminate spurious signals based on consistency thresholds and minimum occurrence requirements. Valid feedback is then routed to appropriate system components—attribution features, optimization criteria, presentation templates, or analysis focus—creating a self-improving cycle where daily usage simultaneously generates value for operators while enhancing system performance through continuous learning.

4 EXPERIMENTAL STUDY

To evaluate Shifu's effectiveness in real-world logistics environments, we designed an experiment that assessed both system performance and learning capability in day to day operational settings. Our approach focused on measuring content quality, feedback incorporation, and continuous improvement through direct operational deployment.

4.1 Experiment Setup

We conducted a two-week proof-of-concept (PoC) across multiple logistics stations in North America during Q1 2025. Test sites were selected based on three criteria: (1) historical high modification patterns when interacting with the predecessor system GLaDOS [9], (2) station managers' and operational experts' (ACES) willingness to participate, and (3) data privacy considerations. This selection strategy ensured we captured stations with diverse operational contexts and engagement patterns.

To maximize learning opportunities, station operators were instructed to complete daily root cause documentation (bridges) even when their stations met performance targets. This approach generated consistent feedback through three channels:

- Content acceptance (preservation of system-generated insights)
- Text modifications (edits to system language or analysis)
- Metric suggestions (operator-identified additional root causes)

Domain experts subsequently validated the quality of operator-modified content through daily email reviews, providing an assessment layer that complemented operational feedback.

4.2 Evaluation Metrics

We developed three complementary metrics to assess Shifu's performance across different dimensions of effectiveness:

4.2.1 *Bridge Quality (BQ).*

BQ measures how effectively the AI-generated content meets user needs with minimal editing required. A score above 80% indicates high-quality content requiring few operator changes. We define *BQ* using a weighted harmonic mean that balances content preservation with modification intensity:

$$BQ = \frac{w_a + w_d + (w_m + w_r)}{\frac{w_d}{\mathcal{P}} + \frac{w_a}{\mathcal{U}} + \frac{w_m+w_r}{\mathcal{I}}}, \tag{1}$$

where:

- w_a, w_d, w_m, and w_r are weights representing the relative importance of additions, deletions, modifications, and reorderings to the original content (set to $w_a = 4$, $w_d = 3$, $w_m = 2$, and $w_r = 1$ in our experiment)
- $\mathcal{P} = \frac{G-D}{G}$ (preservation) measures the percentage of original system-generated content that operators kept
- $\mathcal{U} = \frac{(G-D)}{((G-D)+A)}$ (purity) measures the percentage of the final content that originated from Shifu
- $\mathcal{I} = \max\left(0, \frac{((G-D)-M-R)}{(G-D)}\right)$ (integrity) measures the percentage of kept content that remained unchanged
- G, A, D, M, and R represent the number of sentences that were generated, added, deleted, modified and reordered respectively

This formulation balances the competing priorities of maintaining system-generated content while accommodating necessary operator modifications.

4.2.2 Closed Feedback Loop Effectiveness (CFLE). CFLE evaluates Shifu's ability to successfully incorporate valid operational feedback into future bridge generations. This metric traces the complete feedback journey:

1. Initial operator modification to system-generated content 2. Expert (ACES) validation of the modification's value 3. Successful incorporation into Shifu's knowledge base 4. Application of the learned insight in future analyses

A high *CFLE* score indicates that Shifu effectively captures and applies operational expertise, closing the knowledge acquisition loop between system and operators.

4.2.3 Capacity to Learn (C2L). C2L quantifies Shifu's ability to continuously improve through operational feedback, measuring both:

1. Knowledge acquisition rate: How efficiently the system absorbs new operational insights 2. Application effectiveness: How successfully the system applies learned knowledge

This metric specifically addresses the challenge of learning in environments with limited feedback signals, tracking improvement despite the sparse validation typical in logistics operations.

Detailed formulations for *CFLE* and *C2L* metrics, along with their component calculations and effectiveness thresholds, are presented in the Appendix to maintain focus on experimental outcomes in the main paper.

5 RESULTS

We present findings from our two-week proof-of-concept deployment across five logistics stations, demonstrating Shifu's ability to learn continuously from operational feedback without requiring golden standard examples. Our results show improvements in content quality, effective feedback incorporation, and domain knowledge expansion.

5.1 Bridge Quality

Shifu's Bridge Quality (*BQ*) scores demonstrate a clear performance trajectory across the POC period, as shown in Table 1. Initial *BQ* (73%) improved to consistent performance above our 80% threshold after only 5 days of operational feedback, ultimately stabilizing at 87.9% (σ=6.31%)[1].

Day	*BQ* (%)	σ (%)
1	95	16
2	73	37
3	76	0
4	79	28
5	86	20
6	90	7
7	86	14
8	86	11
9	90	11
10	92	9
11	85	13

Table 1: Bridge Quality scores showing rapid improvement and stabilization above the 80% quality threshold.

The transition from adjustment period (Feb 20-24, avg=76%) to stabilization phase (Feb 25-Mar 5, avg=87.9%) represents an improvement that coincides with our first feedback cycle completion. This confirms our hypothesis that Shifu effectively learns from operational feedback without requiring pre-defined exemplar bridges.

Comparative analysis shows Shifu achieved a 4X improvement over our baseline system. While the production baseline system reported an 80% naive acceptance rate, detailed analysis revealed that 76.5% of its "accepted" insights required substantial modification, with only 5.4% truly accepted without changes (140 of 2625 instances). In contrast, Shifu maintained stable acceptance rates above 85% after just one week of operational learning while testing against stations that frequently modified baseline bridges.

5.2 Feedback Loop Effectiveness

Day	Modified	Verified	Grounded	Closed	Score(%)
1	7	6	3	0	0.00
2	28	22	16	0	0.00
3	32	25	18	1	5.56
4	44	37	27	15	55.56
5	47	39	29	15	51.72
6	54	42	29	15	51.72
7	66	53	31	24	77.42
8	73	57	33	24	72.73
9	79	61	34	24	70.59
10	85	64	35	25	71.43
11	93	68	38	25	65.79
12-16	93	68	38	27	71.05
17	93	68	38	34	89.47

Metric	%
Verified	73.12
Grounded	55.88
Closed	89.47

Table 2: Closed Feedback Loop Effectiveness showing the progression from operator modifications to insights that get validated by ACES team, insights that can be grounded by available observed data, and finally to feedback loop closure. Despite challenging operational conditions, Shifu achieved 89.47% feedback closure.

[1] We dismissed the high acceptance (95%) from the first day as operators were still getting oriented with the new bridge UI and were being reminded about experimental expectations.

Table 2 presents the cumulative results for Shifu's Closed Feedback Loop Effectiveness (*CFLE*). This data reveals two important operational characteristics of Shifu's feedback mechanism.

First, expert validation through email exchanges emerged as a significant bottleneck in the feedback closure process. Contrary to our initial expectations, these exchanges required multiple rounds of clarification to properly map station feedback to specific metrics rather than simple validation. This explains the step-function increases observed on 2/24 (5.56% → 55.56%) and 3/11 (71.05% → 89.47%), which correspond to batch processing of accumulated feedback following extended consultations.

Second, operational time constraints significantly impacted feedback quality. Despite experimental requirements, practical pressures limited operators' ability to provide detailed modification explanations. When encountering unclear content, operators typically deleted problematic sections entirely rather than providing targeted corrections that would yield clearer learning signals. This behavior pattern explains why only 55.9% of ACES-validated modifications could be grounded in accessible operational data.

Despite these challenges, Shifu achieved a final *CFLE* score of 89.47%, exceeding our 80% success threshold. This validates our approach of continuous learning from operational feedback rather than requiring extensive upfront knowledge engineering, as further demonstrated in our knowledge expansion results.

5.3 Knowledge Expansion

Our experiment demonstrated Shifu's ability to expand domain knowledge through operational feedback. We observed knowledge expansion in 4 out of 6 performance categories investigated, with a net addition of 11 new metrics and removal of 2 metrics that proved less relevant in practice.

For the category related to warehouse intake and package scanning timeliness, Shifu expanded the metric coverage by nearly half compared to the baseline system within the initial week of feedback. The system identified particularly valuable additions like Sort Compliance (which appeared across multiple metric categories) and specialized metrics such as Total Volume that Missed Induct Critical Pull Time and Volume Inducted but not Containerized.

Similar expansion patterns occurred across other performance categories:

- "Not Attempted" category: 50% metric increase
- "Not Dispatched" category: 10% metric increase
- "Out of Delivery Time" category: 8% metric increase

This knowledge expansion demonstrates Shifu's ability to adapt to facility-specific operational contexts without requiring comprehensive advance knowledge engineering.

5.4 Facility-Specific Context Adaptation

Our operational deployment revealed distinct patterns in how different facilities interact with Shifu's bridge generation system. Each facility demonstrated unique "signature patterns" in their feedback, reflecting their operational priorities and improvement focus areas:

- **Facility A** consistently prioritized subcontractor management metrics, modifying a substantial majority of bridges to include specific route identifiers and delivery service provider (DSP) performance data.
- **Facility B** focused intensively on internal operations, majority of their modifications adding process stage performance data and personnel metrics.
- **Facility C** displayed a characteristic emphasis on capacity planning, with almost all of their modifications addressing volume management aspects.

Through conversations with domain specialists, we discovered that facilities selectively focus on areas they perceive as their biggest improvement opportunities rather than providing comprehensive on all aspects of operational performance. Intentionally, facilities modify sections most relevant to their core operational focus areas while often leaving other aspects unaddressed, even when those unmodified sections contain potential inaccuracies.

This finding suggests the value of facility-customizable analysis categories. As one expert noted: "I'd like to see AI Insights allow facilities to optimize the categories themselves." This validates our approach of building facility-specific knowledge models rather than attempting to create a universal model that would inadequately address unique operational contexts.

6 CONCLUSION AND FUTURE WORK

Shifu demonstrates a novel approach to knowledge acquisition and root cause analysis in logistics environments, addressing the challenges of documentation quality variance, limited expert review, absence of gold standards, and context specificity. Our experimental results show that Shifu can improve content quality (reaching 87.9% Bridge Quality score within one week), effectively close feedback loops (89.5% closure rate), and expand domain knowledge (44% metric expansion in key categories) without requiring extensive upfront knowledge engineering.

6.1 Limitations

While Shifu shows promising results, several limitations affect its current implementation:

Data Access Constraints: The primary barrier to achieving human-equivalent analysis is data availability rather than reasoning capabilities. As a Process Improvement Specialist noted, "For the new model to be able to explain [issues] as well as an operator can and quicker, it needs Quicksight... I open a couple of Quicksight dashboards and know what happened in the station." Many critical metrics become available only after the analysis deadline, creating a timing mismatch that affects quality. Additionally, some metrics are managed by federated teams who did not enable metric deep dive access during our experiment.

Feedback Quality Dependencies: Shifu's learning rate is constrained by the quality and consistency of user feedback, which varies significantly across stations, operators, and field experts. Our results show that only 55.9% of operator modifications were sufficiently detailed to be grounded in accessible operational data.

Correlation vs. Causality: The current system identifies statistical relationships but may present correlations as causal relationships, potentially leading to ineffective operational interventions. As we advance toward causal inference, robust data collection mechanisms established through Shifu will enable more sophisticated approaches.

Cold Start and Echo Chamber Risks: New metric categories or facilities without historical feedback may experience initially lower bridge quality. Additionally, as Shifu adapts to facility-specific preferences, it risks reinforcing existing operational biases rather than identifying optimal strategies.

6.2 Future Work

Based on our findings, we identify three key directions for future development:

Enhanced Data Integration: Future iterations will focus on improving data accessibility by integrating Quicksight dashboards and other operational data sources that currently inform human analysis. We will work with federated teams to enable deeper metric access while addressing the timing mismatch between data availability and analysis deadlines through predictive approaches.

Production Readiness Improvements: While AutoML (specifically AutoGluon) has proven effective for prototyping, we will implement more controlled approaches for production environments including:

- **Multi-Station Benchmarking:** Testing comprehensive approaches across multiple facilities selected based on field expert proximity or regional director alignment
- **Prompt Engineering Optimization:** Evaluating cost-performance trade-offs with newer models, adapting dynamic exemplar techniques, and applying optimization approaches like COPRO [21] and MIPRO [17]
- **Advanced Explainability:** Testing alternative interpretability methods from the imodels library [18] to enhance understanding of Shifu's insights

Causal Reasoning Development: Building on our current correlation-based approach, we will work toward more sophisticated causal inference methods. The systematic data collection established through Shifu provides the foundation for this transition, potentially addressing the long-standing challenge of separating correlation from causation in logistics performance analysis.

Shifu provides a blueprint for continuous learning systems in domains with subjective quality criteria and specialized operational contexts. By balancing automated analysis with human expertise, this approach offers a path forward for knowledge-intensive applications where neither pure automation nor fully manual processes are optimal.

REFERENCES

[1] Anonymous. [n. d.]. Root Cause Expert System.

[2] Anonymous. [n. d.]. Whispering Angel.

[3] Anonymous. 2025. Annotate-LLM: A General Multi-modal LLM Framework for Annotation Generation in Risk Investigation. In *Proceedings of the 12th Amazon Consumer Science Conference (CSS 2025)*.

[4] Hugo Botelho, Paulo Peças, Diogo Jorge, James Mcleod, Loris Albertoni, Luís Caldas de Oliveira, and Marco Leite. 2024. Data-Driven Root-Cause Analysis in the Scope of Continuous Improvement Projects. In *Advances in Production Management Systems. Production Management Systems for Volatile, Uncertain, Complex, and Ambiguous Environments (IFIP Advances in Information and Communication Technology, Vol. 730)*, Matthias Thürer, Ralph Riedel, Gregor von Cieminski, and David Romero (Eds.). Springer, Cham, 31–45. https://doi.org/10.1007/978-3-031-71629-4_3

[5] Yinfang Chen, Huaibing Xie, Minghua Ma, Yu Kang, Xin Gao, Liu Shi, Yunjie Cao, Xuedong Gao, Hao Fan, Min g Wen, Jun Zeng, Supriyo Ghosh, Xuchao Zhang, Chaoyun Zhang, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Tianyin Xu. 2024. Automatic Root Cause Analysis via Large Language Models for Cloud Incidents. In *Proceedings of the Nineteenth European Conference on Computer Systems (EuroSys '24)*. Association for Computing Machinery. https://doi.org/10.1145/3627703.3629553

[6] Alessandro Chiaraviglio, Sabrina Grimaldi, Giovanni Zenezini, and Carlo Rafele. 2025. Overall Warehouse Effectiveness (OWE): A New Integrated Performance Indicator for Warehouse Operations. *Logistics* 9, 1 (2025), 7. https://doi.org/10.3390/logistics9010007

[7] Ahmed Dawoud and Shravan Talupula. 2025. ProRCA: A Causal Python Package for Actionable Root Cause Analysis in Real-world Business Scenarios. *arXiv preprint arXiv:2503.01475* (Mar 2025). arXiv:2503.01475 [cs.AI] https://arxiv.org/abs/2503.01475

[8] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. 2020. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data. In *Advances in Neural Information Processing Systems*. https://papers.nips.cc/paper/2020/file/55d99b28fe264df581b1cfa949a0cb0d-Paper.pdf

[9] Akash Gupta, P Aditya Sreekar, Sahil Verma, and Abhishek Persad. 2023. Bridging the gap: Empowering Leaders with Automated Narratives Using LLMs.

[10] Ian Hobkirk. 2007. *Technology Strategies for Multi-Site Warehouse and Order Management: Meeting Customer Requirements While Lowering Logistics Costs*. Research Report. Aberdeen Group. A study of 146 multi-site distributors examining challenges and best practices in multi-site warehouse and order management..

[11] Martin Jeske, Martin Wegner, and Markus Kelhaus. 2013. *Big Data in Logistics: A DHL Perspective on How to Move Beyond the Hype*. Trend Report. DHL Customer Solutions & Innovation, Troisdorf, Germany. https://www.dhl.com/content/dam/downloads/g0/about_us/innovation/CSI_Studie_BIG_DATA.pdf Published in cooperation with T-Systems and Detecon Consulting.

[12] Josiane Kroll, Emden Leer, and Manal Assaad. 2016. Challenges and Practices for Effective Knowledge Transfer in Globally Distributed Teams: A Systematic Literature Review. In *Proceedings of the 8th International Conference on Knowledge Management and Information Sharing (KMIS 2016)*. SCITEPRESS - Science and Technology Publications, 49–60. https://doi.org/10.5220/0006045700490060

[13] Chenghao Liu, Wenzhuo Yang, Himanshu Mittal, Manpreet Singh, Doyen Sahoo, and Steven C. H. Hoi. 2023. PyRCA: A Library for Metric-based Root Cause Analysis. *arXiv preprint arXiv:2306.11417* (Jun 2023). https://doi.org/10.48550/arXiv.2306.11417 Github repo: https://github.com/salesforce/PyRCA.

[14] David Maoujoud, Marco Bocchetti, Alexandru Amihalachioaei, Jessy Cyganczuk, Ryan Valoris, Subhadip Duttagupta, Vladyslav Kovchuh, Aurelien Benoit, Joao Patricio, Xiaoyu Mao, Rashmi Singh, and Valentina Cortes Roncagliolo. 2024. On Simplifying Business Review Preparation. In *CSS 2024: Consumer Science Summit*. 13. https://www.amazon.science/#/paper/CSS2024%23679813c0-bec7-4e05-9ad3-933185dbbfe5

[15] Nastaran Okati, Sergio Hernan Garrido Mejia, William Roy Orchard, Patrick Blöbaum, and Dominik Janzing. 2024. Root Cause Analysis of Outliers with Missing Structural Knowledge. *arXiv preprint arXiv:2406.05014* (2024). arXiv:2406.05014 [stat.ML]

[16] Walter J. Ong. 1982. *Orality and Literacy: The Technologizing of the Word*. Methuen, London. Discusses how writing restructures consciousness and serves as an external cognitive tool.

[17] Kawin Opsahl-Ong, Matthew J. Ryan, Joseph Purtell, David Broman, Christopher Potts, Matei Zaharia, and Omar Khattab. 2024. Optimizing instructions and demonstrations for multi-stage language model programs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Miami, Florida, USA, 9340–9366. https://doi.org/10.18653/v1/2024.emnlp-main.525

[18] Chandan Singh, Keyan Nasseri, Yan Shuo Tan, Tiffany Tang, and Bin Yu. 2021. *imodels: a python package for fitting interpretable models*. https://doi.org/10.21105/joss.03192

[19] Akanksha Tirthgirikar. [n. d.]. [Whispering Angel] Generating insights through MLOps pipeline.

[20] Thorsten Wittkopp, Philipp Wiesner, and Odej Kao. 2024. LogRCA: Log-based Root Cause Analysis for Distributed Services. arXiv:2405.13599 [cs.LG] https://arxiv.org/abs/2405.13599

[21] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc Le, Denny Zhou, and Xi Chen. 2024. Large language models as optimizers. *arXiv preprint arXiv:2309.03409* (2024). https://arxiv.org/abs/2309.03409

[22] Michael Zimmerman, Josh Brogan, Balika Sontalia, and Korhan Acar. 2023. *The 34th Annual State of Logistics Report: The Great Reset*. Annual Report. Council of Supply Chain Management Professionals, Lombard, Illinois. https://cscmp.org/store/detail.aspx?id=SOLR23 Produced in partnership with Kearney and Penske Logistics.

A DETAILED EVALUATION METRICS

This appendix provides detailed formulations for the Closed Feedback Loop Effectiveness (CFLE) and Capacity to Learn (C2L) metrics referenced in Section 4.

A.1 Closed Feedback Loop Effectiveness (CFLE)

CFLE measures Shifu's ability to successfully incorporate valid operational feedback into future bridge generations. It specifically quantifies the percentage of operator modifications that complete the full feedback cycle—from initial modification to system incorporation.

A.1.1 Definition. CFLE is defined as the percentage of closed feedback loops out of the total correct and closable feedback loops. A feedback loop is considered closable when the modifications made by a station meet two criteria:

(1) They are verified by operational experts (ACES)
(2) They are grounded in data accessible to the Shifu system

Success is defined as achieving a score exceeding 80% of feedback loops closed out of total possibly closable feedback loops.

A.1.2 Calculation Methodology. The CFLE calculation follows a simple three-step process:

Step 1: Identify and categorize all operator modifications
We track four types of modifications made by operators:

- Content additions (new insights)
- Content deletions (removing irrelevant insights)
- Content modifications (changing existing insights)
- Content reordering (changing priority sequence)

Step 2: Determine valid closable feedback loops
For each modification, we check:

- Is it verified by operational experts (ACES)?
- Is it grounded in data accessible to Shifu?

Only modifications that meet both criteria are counted as valid closable feedback loops.

Step 3: Calculate closure rate
The CFLE score is calculated using a simple percentage:

$$\text{CFLE Score} = \frac{\text{Number of successfully closed feedback loops}}{\text{Number of valid closable feedback loops}} \times 100\%$$

Where:

- **Successfully closed feedback loops** means the system has incorporated the feedback in subsequent analyses
- **Valid closable feedback loops** means the total number of feedback items that meet the verification and data grounding criteria

A.1.3 Data Sources. CFLE calculation relies on three primary data sources:

- **ACES Verification**: Daily expert validation of operator modifications through structured email exchanges
- **Data Grounding Assessment**: Verification that modifications reference metrics accessible to Shifu
- **Closure Confirmation**: Re-running analysis to confirm that feedback has been incorporated

A.2 Capacity to Learn (C2L)

C2L quantifies Shifu's ability to continuously improve through operational feedback, focusing specifically on the relative velocity of improvement between content quality and feedback incorporation.

A.2.1 Definition. C2L is defined as the average time taken to close valid feedback loops, with a target of being faster than the time required for bridge quality improvements. This metric is particularly important when the system demonstrates suboptimal initial bridge quality (60%–80%), as it indicates how quickly the system can adapt to operational needs.

Success is defined as achieving a feedback loop closure velocity that equals or exceeds the velocity of bridge quality improvement.

A.2.2 Calculation Methodology. The C2L calculation involves comparing two different velocity measurements:

Step 1: Track daily scores for both metrics

- Bridge Quality Score: How well the system content meets user needs
- Feedback Loop Score: How effectively the system closes feedback loops

Step 2: Calculate daily changes
For each day after day 1, we calculate the day-to-day change in scores:

$$\Delta\text{Quality}_{\text{day}} = \text{QualityScore}_{\text{day}} - \text{QualityScore}_{\text{day-1}}$$

$$\Delta\text{Feedback}_{\text{day}} = \text{FeedbackScore}_{\text{day}} - \text{FeedbackScore}_{\text{day-1}}$$

Step 3: Calculate average velocity
Velocity is simply the average rate of change across all days of the evaluation:

$$\text{Velocity}_{\text{Quality}} = \frac{\text{Sum of all daily quality changes}}{\text{Number of days - 1}}$$

$$\text{Velocity}_{\text{Feedback}} = \frac{\text{Sum of all daily feedback changes}}{\text{Number of days - 1}}$$

Step 4: Compare velocities
Success is achieved when the feedback velocity is greater than or equal to the quality velocity:

$$\text{Velocity}_{\text{Feedback}} \geq \text{Velocity}_{\text{Quality}}$$

This indicates that the system is learning and incorporating feedback at a pace that keeps up with or exceeds improvements in content quality.

A.2.3 Interpretation. The C2L metric helps us understand whether:

- The system learns quickly enough to drive quality improvements
- Improvements in quality are coming from feedback incorporation
- The feedback loop is functioning effectively as a learning mechanism

A positive quality velocity with a comparable or higher feedback velocity suggests an effective continuous learning system.

A.2.4 Data Sources. C2L calculation relies on daily measurements of:

- **Bridge Quality Scores**: Derived from the Bridge Quality metric as defined in Section 4
- **Feedback Loop Scores**: The daily CFLE scores as described above

The velocity measurements provide insight into the system's learning dynamics, with an optimal result showing bridge quality improvements that match or exceed the pace of feedback incorporation.