Trustworthy LLM-Based Rule Learning for Auditable Agentic Decision-Making

Jie Zhou
Amazon
Seattle, WA, USA
jiezhoua@amazon.com

Zhenyu Zhang
Amazon
Seattle, WA, USA
zhenyuzh@amazon.com

Ted Sandler
Amazon
Seattle, WA, USA
sandler@amazon.com

Xiaolong Kuang
Amazon
Seattle, WA, USA
xkkuang@amazon.com

Xinyang Sheng
Amazon
Seattle, WA, USA
xinyans@amazon.com

Ozalp Ozer
Amazon
Seattle, WA, USA
ozalpo@amazon.com

ABSTRACT

Deploying LLM-based agentic systems in high-stakes operational settings requires achieving trustworthiness, explainability, consistency, and adaptability simultaneously. Existing approaches such as manual expert audit, supervised classifiers, or hard-coded rule systems struggle to achieve them within a unified framework. In this paper, we introduce a LLM-based rule learning framework comprising three modules: (1) an error-driven rule induction algorithm that iteratively extracts both explicit and implicit domain knowledge from limited labeled data via Chain-of-Thought prompting, with a greedy set-cover-based pruning step to select a minimal, high-utility rule set; (2) a rule deduction module that generates predictions using specific rules with auditable reasoning traces for each decision; and (3) a rule generalization module that autonomously infers new rules from out-of-distribution data. We evaluate the framework on a real-world record reconciliation task. The system achieves 92% reconciliation accuracy compared to 85% for a hard-coded rule-based baseline. Ablation studies confirm that the induced rule library acts as a consistency anchor, improving accuracy from 84% to 92%. The rule generalization module identifies valid new rules from cases that can't be handled by existing rule library in 50% of samples, demonstrating the system's adaptability to evolving operational environment. We discuss evaluation challenges unique to production agentic systems, including sensitivity to data representation, the trade-off between rule quantity and quality, and strategies for continuous monitoring under absent ground truth.

KEYWORDS

agentic AI evaluation, LLM trustworthiness, rule induction, human-in-the-loop, distribution shift, explainability

ACM Reference Format:
Jie Zhou, Zhenyu Zhang, Ted Sandler, Xiaolong Kuang, Xinyang Sheng, and Ozalp Ozer. 2026. Trustworthy LLM-Based Rule Learning for Auditable Agentic Decision-Making. In *Proceedings of KDD Workshop on Evaluation and Trustworthiness of Agentic AI (KDD'26).* ACM, New York, NY, USA, 7 pages. https://doi.org/XXXXXXX.XXXXXXX

KDD'26, August 2026, Jeju, Korea

ACM ISBN 978-1-4503-XXXX-X/2018/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

As LLM-based agentic systems are increasingly deployed in production environments for complex decision-making, evaluating their trustworthiness becomes a critical challenge [1]. Unlike traditional ML models evaluated on static benchmarks, agentic systems exhibit stochastic behavior, operate in evolving environments, and must provide explainable decisions—particularly in high-stakes domains where errors carry significant financial consequences. The prevalence of domain-specific data further compounds this difficulty, as the implicit knowledge required to judge correctness is often scattered across systems.

In practice, several strategies exist for monitoring and evaluating such systems, each with fundamental limitations. First, domain associates periodically audit decisions to assess system performance. While reliable, this approach is costly, slow, and does not scale to the volume of decisions made by production systems. Second, lightweight classifiers trained on annotated data can automate performance monitoring. However, these models require large-scale labeled data that is expensive to obtain and need to be retrained when the underlying data distribution shifts. Third, domain experts abstract recurring patterns from audit judgements and formalize them into standard operating procedures (SOPs) or rules that can be applied systematically. While rule-based systems offer high interpretability and deterministic decisions [3], they are inherently rigid. They fail on cases that deviate even slightly from predefined criteria and cannot generalize to unseen cases outside predefined rules.

To address these, we propose an LLM-based agentic framework that combines rule induction, deduction, and generalization with iterative human-in-the-loop rule refinement to achieve accuracy, explainability and adaptability simultaneously. In the rule induction stage, the system builds a rule library from limited labeled data, capturing both explicit rules commonly used in practice and implicit rules embedded in domain expertise. Since operational patterns evolve over time, the rule generalization module automatically infers new rules when encountering data with previously unseen patterns, addressing the fragility that prior work has identified in LLM inductive reasoning under distributional noise [5]. In the

rule deduction stage, the system applies the learned rules to solve reasoning problems while providing clear explanations grounded in specific rules for each decision.

Critically, we frame our contribution not merely as a system paper, but as an *evaluation framework* that addresses multiple dimensions of agentic AI trustworthiness:

- **Decision accuracy**: The LLM generates and verifies rules over a small annotated set, extracting both explicit and implicit domain knowledge. This reduces the hallucination risk inherent in prompting approaches that rely solely on the model's embedded knowledge, while also eliminating dependence on manually crafted rules that may not be directly interpretable by the model.
- **Explainability**: The induced rule library serves as a centralized, human-readable knowledge base that mirrors how domain experts reason from established guidelines. Rather than relying on opaque internal representations, the system grounds each decision in specific rules drawn from this library, providing transparent reasoning traces that can be audited and verified.
- **Robustness to distribution shift**: We evaluate the agent's ability to generalize to novel patterns/rules through a rule generalization module, testing whether the system can autonomously identify valid new rules when encountering previously unseen data distributions.
- **Human-in-the-loop governance**: We design an iterative evaluation loop where domain experts validate agent outputs, correct errors, and feed validated labels back into the system, establishing a continuous monitoring and improvement cycle.

We demonstrate system efficacy on an industrial inventory matching use case where shortage inventory must be reconciled with surplus inventory to reduce defects. This domain exemplifies the challenges motivating our work: supervised models trained on matching labels require frequent retraining, ground truth is expensive to obtain, and model decisions are difficult to interpret and evaluate. Domain specialists can investigate cases using their expertise but cannot scale. An agentic system must therefore act with domain-expert-level knowledge, adapt to new patterns, and provide auditable explanations for each decision. Our experiments show that the agentic system achieves 92% reconciliation accuracy on 4K test cases, outperforming the deterministic baseline of 85%. Ablation studies reveal that without the rule induction module, accuracy drops to 84%, confirming that induced rules mitigate the inherent stochasticity of LLM generation. The rule generalization module successfully identifies valid new rules from unlabeled data, demonstrating the system's adaptability without retraining.

The remainder of this paper is organized as follows. Section 2 reviews related work on LLM evaluation and rule learning. Section 3 describes the agentic system architecture. Section 4 presents the dataset and trustworthiness evaluation across multiple dimensions. Section 5 discusses evaluation challenges and lessons learned. Finally, Section 6 concludes with future directions for agentic AI evaluation.

2 RELATED WORK

Evaluation of Agentic AI Systems. As LLM-based agents are deployed in production, evaluation has emerged as a critical research area. Traditional benchmarks assess static capabilities, but agentic systems require evaluation of multi-step reasoning, adaptability, and behavioral consistency over time. Recent work highlights challenges including stochastic agent behavior, absence of ground truth, and the need for continuous post-deployment monitoring. LLM-as-judge methods have been proposed for scalable evaluation, though they introduce their own biases. Our work contributes to this space by proposing an evaluation framework that combines automated metrics with human-in-the-loop validation for a production agentic system.

Trustworthiness and Explainability. Trustworthy AI requires systems that are reliable, fair, robust, and explainable. For LLM-based agents, explainability is particularly challenging due to the opacity of neural generation. Rule-based explanations offer one path to interpretability—if the agent can articulate which rules it applied and why, human auditors can verify its reasoning. Our framework evaluates explainability through induced rule libraries that serve as both decision logic and audit trails.

LLM Rule Learning. Recent studies have integrated LLMs with rule learning to improve reasoning and generalization. [10] integrates LLMs with rule-based systems for interpretable autonomous driving. [14] leverages LLMs to learn rules from robot trajectories, refine them, and translate to code. [13] explored LLM performance in following predefined rules. [11] leverages LLMs to infer logic rules from training data for downstream reasoning. [6] use LLM generalization to generate new detection rules for web security. The most relevant work is [15] which prompts LLMs to learn rule libraries for reasoning. However, none of these works evaluate the trustworthiness dimensions of rule-learning agents—explainability, robustness to distribution shift, and human-in-the-loop governance—which is the focus of our contribution.

In-Context Learning and Chain-of-Thought. LLMs with hundreds of billions of parameters have demonstrated remarkable emergent capabilities. [1] introduced *in-context learning*, where LLMs generate expected outputs from few examples without gradient updates. Instruction tuning enables generalization to unseen tasks [7, 8]. Chain-of-Thought (CoT) prompting [9] prompts LLMs to reason step-by-step, improving performance on complex tasks. These capabilities underpin our agentic system's ability to induce and apply rules, but also introduce evaluation challenges around consistency and reliability that we address in this work.

3 METHODS

3.1 System Overview

As shown in Figure 1, the agentic system consists of two main modules: **rule induction** and **rule deduction**. During induction, Given a small set of human-annotated examples, the LLM generates candidate rules that explain observed decision patterns, then prunes redundant or contradictory candidates through verification against the labeled data. The resulting *rule library* captures both explicit criteria and implicit domain knowledge that experts apply intuitively but rarely formalize. During deduction, the agent retrieves applicable rules from the library, applies them to the input data,

and produces a prediction along with an explanation trace specifying which rules were triggered and why. When the deduction module encounters cases that cannot be resolved by existing rules, the system routes them to a generalization module. Here, the LLM leverages the existing rule library as context alongside the novel data to hypothesize new rules. Newly proposed rules are routed to human validation before integration into the library.

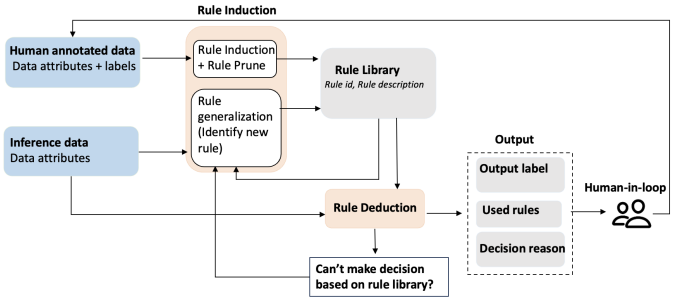

Figure 1: Agentic system architecture with evaluation touchpoints

3.2 Rule Induction

The rule induction module extracts structured decision rules from a small set of curated labeled data, constructing a centralized rule library that serves as the knowledge base for downstream reasoning. Our method operates in a low-data regime, making it practical for domains where labeling is expensive and expert time is scarce. The induction process follows a two-stage algorithm (Algorithm 1): *iterative rule induction* to maximize rule coverage, followed by *rule pruning* to eliminate redundancy and retain only high-utility rules.

Algorithm 1 Two-stage Rule Induction with CoT

Require:
  Induction data $I = \{(\mathbf{X}_i, l_i)\}_{i=1}^{N}$
  Validation data $V = \{(\mathbf{X}_j, l_j)\}_{j=1}^{M}$
Step 1: Iterative Rule Induction
Initialize error set $E = I$
$k = 0, R = \emptyset$
while $E \neq \emptyset$ AND $k < K$ **do**
    $k = k + 1$
    Run CoT induction on E to induce rules R_k
    Perform deduction on E to predict label $\hat{l}_i$ using E and R
    $E = \{(\mathbf{X}_i, l_i) \mid \hat{l}_i \neq l_i\}_{i=1}^{N}$
    $R = R \cup R_k$
end while
Step 2: Rule Pruning
Apply Algorithm 2 on $R = \{R_1, \ldots, R_k\}$ and V
Select optimal rule set R^* maximizing accuracy on V
return Optimal rule set R^*

In Step 1, given induction data I where $\mathbf{X}_i$ denotes the attribute set and l_i the ground-truth label and the validation data V, we formulate the rule induction as an iterative hypothesis-and-verify process. At each iteration k, the LLM is prompted with Chain-of-Thought (CoT) reasoning over the current error set E to hypothesize a batch of candidate rules R_k that explain the observed label patterns. The combined rule set $R \cup R_k$ is applied via the deduction module (Section 3.3) to predict labels $\hat{l}_i$ for all samples in E. The error set is then updated to retain only misclassified samples. Iteration continues until the error set is empty ($E = \emptyset$) or the maximum iteration count K is reached. Here, K is selected based on heuristics (can be size of induction data N). This error-driven loop is analogous to boosting in ensemble learning: each iteration focuses the LLM's inductive attention on residual errors that prior rules failed to capture, progressively extracting rules that cover common patterns first and edge cases later.

The iterative induction process intentionally over-generates rules to maximize recall over diverse patterns. However, redundant or overlapping rules degrade inference quality for two reasons: (1) the LLM's attention mechanism may over-weight repeated instructions at the expense of other critical rules during deduction, and (2) a bloated rule library increases prompt length and inference cost without much accuracy gains.

We address this through a greedy set-cover formulation inspired by the weighted maximum coverage problem [14]. As shown in Algorithm 2, given the validation set V and the candidate rule library $R = \{R_1, \ldots, R_K\}$, we seek a minimal subset $R^* \subseteq R$ that maximizes prediction accuracy on V. We define the *marginal cover rate* of a candidate rule R_i with respect to the current mislabeled set W as:

$$\text{Cover Rate} = \frac{\text{Number of mislabeled cases corrected by the rule}}{\text{Total number of mislabeled cases}}.$$

At each iteration, the rule with the highest cover rate is added to our refined rule library R^*, and W is updated to reflect remaining errors. This process continues until no additional rules improve coverage.

Algorithm 2 Rule Prune Algorithm

Require:
  Sample set $V = \{(\mathbf{X}_j, l_j)\}_{j=1}^{M}$
  Rule library $R = \{R_1, ..., R_K\}$
  Initialize optimal rule set $R^* = \{R_i$ with highest accuracy$\}$
  Initialize cover rate $CR = 1$
  Initialize mislabeled data $W = \{(\mathbf{X}_j, l_j)\}_{j=1}^{M}$
while $CR > 0$ **do**
    $R_{rest} = R - R^*$
    for $R_i \in R_{rest}$ **do**
      Perform deduction on W using $R^* \cup R_i$
      Calculate cover rate CR_i
    end for
    Select R_i^* with highest $CR_i^* = \max_i CR_i$
    $R^* = R^* \cup R_i^*$
    Update $W = \{(\mathbf{X}_i, l_i) \mid \hat{l}_i \neq l_i\}_{i=1}^{M}$
end while
return Optimal rule set R^*

To adapt to emerging new patterns, we introduce a rule generalization module. When the deduction module cannot reach a

decision using the existing rule library, the generalization module is triggered. The LLM is prompted to infer new rules for unlabeled or limited labeled input data based on the current rule library and high-level domain knowledge. This design is grounded in two theoretical observations. First, LLMs encode abstract relational structures that transfer across tasks and distributions [1, 4] –they learn general types of relationships rather than specific feature combinations [12], enabling analogical reasoning over unseen data. Second, grounding hypothesis generation in an established rule library constrains the search space, mitigating the hypothesis drift and pattern overfitting observed in unconstrained LLM inductive reasoning [5]. Newly inferred rules are routed to human experts for validation, preserving its integrity while allowing the system to evolve with shifting data distributions.

3.3 Rule Deduction

During the deduction stage, the LLM outputs the predicted label, the specific rules it applied, and a natural-language reasoning trace behind model decision. Downstream systems consume the prediction for execution while surfacing the reasoning trace as a stakeholder-facing explanation. From an evaluation standpoint, it enables auditors to verify not just *what* the agent decided, but *why*. In practice, we regularly sample and evaluate model outputs to validate its decisions. Human validated labels let us update the rule library by running induction on the expanded labels.

4 EXPERIMENTS

4.1 Data

We evaluate our system on a discrepancy resolution task in industry. Shortage record (expected but missing inventory) and surplus (unexpected or excess inventory) are logged in the system. Due to workflow complexity, the true linkage between corresponding shortage and surplus records is often lost—yet accurate matching is required both for financial reporting and for providing stakeholder-facing explanations that justify each reconciliation decision.

Currently, domain specialists manually investigate each shortage case, examining attributes across dozens of surplus candidates to identify the best match. While reliable, this approach does not scale. The existing automated evaluation system relies on hard-coded rules commonly used by domain experts, yet these rules are rigid with predefined criteria. Therefore, an agentic rule learning and evaluation system is needed.

We sample 5K inventory discrepancy cases, collecting attributes for both shortage and surplus records. Each shortage case has approximately 70 candidate surplus matches on average, making the selection space large and accurate matching non-trivial. We split the data into induction set (used for rule learning), validation set (used for rule pruning), and 4K test samples for final evaluation.

For each sample, the shortage and surplus records are described by the following attribute format:

```
<shortage> The missing date is XX, missing quantity is XX,
owner is XX, product is XX,  etc </Missing>
<surplus> ID 1: the found date is XX and hour is XX,
Found quantity is XX, product is XX, etc </Found>
```

Ground-truth labels are costly, requiring expert manual investigation by checking various data. We adopt a disagreement-based annotation strategy: a few-shot LLM and the deterministic rule baseline are independently applied to all testing samples, and only cases where the two disagree are routed to domain experts for annotation. Agreement cases are accepted as pseudo-labels. This focuses annotation effort on the most ambiguous and informative instances.

4.2 Evaluation Framework Design

Evaluating an LLM-based agentic system in production requires addressing challenges absent from traditional ML evaluation. We organize our evaluation along five dimensions grounded in our experimental design as summarized in Table 1. For *accuracy*, we measure reconciliation accuracy as the percentage of correct matches among all reconciled cases. For *coverage*, we report binary label recall, i.e., the percentage of reconcilable cases the agent correctly identifies as reconcilable. For *explainability*, we manually verify each induced rule's validity. For *consistency*, we conduct ablation studies comparing rule-guided vs. direct prompting to quantify how the rule library reduces stochastic variation. For *adaptability*, we evaluate new rule identification rate on unseen dataset.

Table 1: Evaluation dimensions, metrics, and methods used in this work.

Dimension	Metric	Method
Accuracy	Reconciliation accuracy	Comparison with validated labels
Coverage	Binary label recall	Detection of reconcilable cases
Consistency	Accuracy with/without rules	Ablation study
Adaptability	New rule identification rate	Distribution shift test
Explainability	Rule validity	Expert audit of induced rules

4.3 Evaluation Results

Prompting strategy evaluation. In the induction stage, we evaluated standard zero-shot prompting, few-shot prompting, and Chain-of-Thought (CoT) prompting on data labeled by hard-coded rule-based baseline. Few-shot achieved comparable performance to standard prompting (84% reconciliation accuracy). CoT prompting adds a deep analysis module that prompts the LLM to generate internal reasoning traces before articulating rules. CoT achieved significantly higher reconciliation accuracy (92% vs 84%) and generated a more complete and precise rule library compared with standard prompting. Additionally, CoT identified additional rules not identified by standard prompting strategy.

Data representation evaluation. Converting tabular attribute data into LLM-readable format significantly impacts agent performance. We evaluated three representation strategies—JSON, markdown tables, and textual enumeration—which are widely used in LLM tabular data applications [2]. Table 2 shows that textual enumeration achieves the highest reconciliation accuracy (78%), outperforming JSON (74%) and markdown (32%). The intuition is that textual enumeration converts tabular data into a language-like format that LLMs can naturally parse and reason over, consistent with their pretraining distribution. Further improvements come from using comparison-based features (checking whether surplus attributes match shortage attributes, improving to 80%) and grouping candidates by features (improving to 84%).

Table 2: Reconciliation accuracy across data representation strategies on the validation dataset.

Data representation	Accuracy
JSON	74%
Markdown	32%
Text enumeration	78%
Text enum + Comparison features	80%
Text enum + Comparison + Grouping	84%

Comparison with deterministic baseline. We performed rule induction on curated data to build a rule library and ran the deduction module on test data. The baseline is a deterministic rule-based system built by domain experts who manually identified high-precision reconciliation rules and hard-coded them over the course of many months. These rules tend to be high-level and often fail when multiple attributes must be analyzed simultaneously. We use aforementioned labeling strategy to correct labels from baseline method and report final evaluation results in Table 3. From the table, the agentic system achieves 92% overall accuracy vs. 85% for the deterministic baseline and comparable recall.

Table 3: Evaluation of reconciliation accuracy and binary label recall: deterministic baseline vs. agentic system.

Size	Accuracy		Recall	
	Rule-based	Agentic	Rule-based	Agentic
4,101	85%	92%	100%	98%

Ablation study: evaluating consistency. To validate the necessity of the rule induction module for behavioral consistency, we conducted an ablation study by removing it entirely and relying solely on zero/few-shot prompting for direct deduction. Without the rule library, accuracy drops to 84%. This demonstrates a key evaluation insight: induced rules help eliminate the inherent stochasticity of LLM generation. When using direct few-shot prompting, LLMs produce inconsistent outputs for similar inputs due to their probabilistic nature. The rule library acts as a *consistency anchor*, ensuring stable reasoning across predictions—a critical property for trustworthy deployment.

Evaluating robustness to distribution shift. A key trustworthiness requirement is the agent's ability to handle novel patterns without retraining. We evaluated this through the rule generalization module on 100 cases from a different data distribution containing a novel feature not present in the training data. The existing rule library was insufficient for these cases.

The rule generalization module automatically proposed new reconciliation rules when existing ones didn't apply. The agent successfully identified a common matching pattern in approximately 50% of cases. This demonstrates that the agent can identify valid reconciliation patterns even when new features are introduced and no labeled data is available. To illustrate the agent's reasoning during rule generalization, we show a representative output below. The agent applies existing rules (quantity match, date match, product match, time proximity) while simultaneously proposing a new container-matching rule:

```
<best_found_id>ID 1</best_found_id>
<used_rules>1, 3, 4, 5, 8, 10</used_rules>
<new_rule>When a Found item has the same container as
the Missing item, it should be preferred over
items in different container, especially when other
key attributes match.</new_rule>
<reconciliation_analysis>
...
</reconciliation_analysis>
```

This output demonstrates three evaluation-relevant properties: (1) the agent provides a complete reasoning trace that can be audited; (2) it correctly applies existing rules while identifying gaps; and (3) the proposed new rule is specific and actionable, enabling human validators to quickly assess its validity.

Evaluating explainability quality. Beyond accuracy metrics, we evaluated the quality of the agent's explanations. For each of the rules in the induced rule library, domain experts verified rule validity. All CoT-induced rules were confirmed as valid domain knowledge.

5 DISCUSSION: EVALUATION CHALLENGES FOR AGENTIC AI

Our experience evaluating this agentic system surfaces several challenges relevant to the broader agentic AI evaluation community:

Absence of ground truth. In production settings, obtaining ground truth labels is prohibitively expensive. We addressed this through a disagreement-driven labeling approach: using high- confidence deterministic outputs as pseudo-labels and targeted human review for cases where the agentic system disagrees with the baseline. This focuses expensive annotation on the most informative cases. However, this approach introduces a potential bias: we may underestimate errors on cases where both systems agree incorrectly. Periodic random audits are necessary to detect such systematic blind spots.

Stochastic agent behavior. LLMs are inherently probabilistic, producing different outputs for identical inputs across runs. Our ablation study quantifies this effect: without the rule library, accuracy drops from 92% to 84%. The rule library serves as a consistency mechanism, constraining the agent's reasoning space to produce stable outputs. This finding has broader implications for agentic AI evaluation: any evaluation framework must account for output variance and measure not just mean performance but consistency across runs.

Sensitivity to input representation. Our data representation evaluation (Table 2) reveals that the same agent with the same rules produces dramatically different accuracy depending on how tabular data is formatted—from 32% (markdown) to 84% (text enumeration with grouping). This sensitivity is a trustworthiness concern: small changes in the input pipeline can cause large performance swings. Evaluation frameworks for agentic systems should include robustness testing across input format variations.

Rule quality vs. rule quantity. The rule pruning algorithm (Algorithm 2) addresses a counterintuitive finding: more rules do not always improve performance. Redundant or repetitive rules

can degrade accuracy because LLMs' attention mechanism may give disproportionate weight to frequently mentioned instructions, causing it to overlook other important rules. Our pruning algorithm, framed as a maximum coverage problem, selects a minimal effective rule set. This suggests that evaluation of rule-learning agents should assess not just rule validity but also rule set optimality.

Evolving data distributions. Production data distributions shift over time as new defect patterns emerge. Static evaluation on fixed test sets cannot capture this. Our rule generalization evaluation provides a methodology for assessing adaptability: hold out data with novel features and measure whether the agent can autonomously discover valid rules. In our experiment, the agent identified valid new rules in 50% of cases from a shifted distribution—useful but not sufficient for fully autonomous operation, reinforcing the need for human-in-the-loop governance.

Explainability enables efficient human oversight. The three-part output structure (prediction, rules used, reasoning trace) enables a form of evaluation that is impossible with black-box models: auditors can identify *why* the agent made an error, not just *that* it erred. In our complex cases, manual review of disagreements between the agentic system and the baseline system revealed that many baseline labels were actually incorrect. The agentic system's reasoning traces facilitated this review process by providing auditors with the agent's rationale for each disagreement.

CoT as an evaluation tool. Our comparison of standard vs. CoT prompting reveals that CoT serves dual purposes: it improves agent performance (92% vs 84% accuracy) and simultaneously produces richer evaluation artifacts. The reasoning traces generated during CoT induction expose the agent's understanding of domain relationships, enabling qualitative assessment of whether the agent has correctly internalized domain knowledge.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we present an evaluation framework for an LLM-based agentic system applied to record reconciliation, addressing multiple dimensions of trustworthiness: explainability through induced rule libraries, consistency through rule-constrained generation, robustness to distribution shift through autonomous rule generalization, and governance through human-in-the-loop iterative refinement. The agentic system achieves 92% accuracy on 4K test cases while providing transparent reasoning traces, and demonstrates adaptability to novel patterns without retraining.

Our findings suggest several directions for future work in agentic AI evaluation:

- **Continuous monitoring metrics**: Developing real-time metrics for detecting performance degradation as data distributions evolve, including rule coverage drift and confidence calibration.
- **Scalable human oversight**: Designing efficient audit protocols that maximize evaluation coverage while minimizing expert annotation burden, potentially using LLM-as-judge methods for preliminary screening.
- **Cross-domain generalization**: Evaluating whether the rule induction–deduction–generalization framework transfers to other high-stakes agentic applications (e.g., fraud detection, compliance monitoring) with similar trustworthiness requirements.
- **Behavioral consistency benchmarks**: Developing standardized benchmarks for measuring LLM agent consistency across runs, including sensitivity to prompt variations and input ordering.

REFERENCES

[1] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. , 25 pages.

[2] Xi Fang, Weijie Xu, Fiona Anting Tan, Ziqing Hu, Jiani Zhang, Yanjun Qi, Srinivasan H. Sengamedu, and Christos Faloutsos. 2024. Large Language Models (LLMs) on Tabular Data: Prediction, Generation, and Understanding - A Survey. *Transactions on Machine Learning Research* (2024). https://openreview.net/forum?id=IZnrCGF9WI

[3] Giovanni De Gasperis and Sante Dino Facchini. 2025. A Comparative Study of Rule-Based and Data-Driven Approaches in Industrial Monitoring. *ArXiv* abs/2509.15848 (2025). https://api.semanticscholar.org/CorpusID:281410785

[4] Or Honovich, Uri Shaham, Samuel Bowman, and Omer Levy. 2023. Instruction Induction: From Few Examples to Natural Language Task Descriptions. *Association for Computational Linguistics* (2023), 1935–1952.

[5] Chunyang Li, Weiqi Wang, Tianshi Zheng, and Yangqiu Song. 2025. Patterns Over Principles: The Fragility of Inductive Reasoning in LLMs under Noisy Observations. (2025), 19608–19626. https://doi.org/10.18653/v1/2025.findings-acl.1006

[6] Jinghua Liu, Yi Yang, Kai Chen, and Miaoqian Lin. 2025. Generating API Parameter Security Rules with LLM for API Misuse Detection. *arxiv* (01 2025).

[7] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '22)*. Curran Associates Inc., Red Hook, NY, USA, Article 2011, 15 pages.

[8] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng-Xin Yong, Harshit Pandey, Michael Mckenna, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *ICLR 2022 - Tenth International Conference on Learning Representations*. Online, Unknown Region. https://inria.hal.science/hal-03540072

[9] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '22)*. Curran Associates Inc., Red Hook, NY, USA, Article 1800, 14 pages.

[10] Fanzhi Zeng, Siqi Wang, Chuzhao Zhu, and Li Li. 2025. ADRD: LLM-Driven Autonomous Driving Based on Rule-based Decision Systems. *arXiv* (2025).

[11] Yudi Zhang, Pei Xiao, Lu Wang, Chaoyun Zhang, Meng Fang, Yali Du, Yevgeniy S. Puzyrev, Randolph Yao, Si Qin, Qingwei Lin, Mykola Pechenizkiy, Dongmei Zhang, S. Rajmohan, and Qi Zhang. 2025. RuAG: Learned-rule-augmented Generation for Large Language Models. *ICLR* abs/2411.03349 (2025). https://api.semanticscholar.org/CorpusID:273850253

[12] Denny Zhou, Nathanael Scharli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed H. Chi. 2023. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. *ICLR* abs/2205.10625 (2023). https://api.semanticscholar.org/CorpusID:248986239

[13] Ruiwen Zhou, Wenyue Hua, Liangming Pan, Sitao Cheng, Xiaobao Wu, En Yu, and William Yang Wang. 2025. RuleArena: A Benchmark for Rule-Guided Reasoning with LLMs in Real-World Scenarios. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (Eds.). Association for Computational Linguistics, Vienna, Austria, 550–572. https://doi.org/10.18653/v1/2025.acl-long.27

[14] Siyu Zhou, Tianyi Zhou, Yijun Yang, Guodong Long, Deheng Ye, Jing Jiang, and Chengqi Zhang. 2025. WALL-E: World Alignment by Rule Learning Improves World Model-based LLM Agents. *NeurIPS* (2025).
[15] Zhaocheng Zhu, Yuan Xue, Xinyun Chen, Denny Zhou, Jian Tang, Dale Schuurmans, and Hanjun Dai. 2024. Large Language Models can Learn Rules. *ICLR* (2024). https://openreview.net/forum?id=tAmfM1sORP

A DATA PREPROCESSING

The data representation comparison and CoT prompting evaluation are presented in Section 4.2 of the main text. Here we provide additional examples of the data formats evaluated.

For JSON format, each row of tabular data is represented as a dictionary:

```
<shortage>{'attr_1': XX, 'attr_2': XX,
'attr_3': XX, ..., 'attr_n': 'XX',}</shortage>
<surplus>{'id': 1, 'attr_1': XX, 'attr_2': XX,
'attr_3': XX, ..., 'attr_n': XX}</surplus>
```

For markdown format, the entire table is inserted directly:

```
Missing table:
| attr_1 | attr_2 | attr_3 | ... | attr_n |
| --- | --- | --- | --- | --- |
| XX | XX| XX | XX | XX |

Found table:
| attr_1 | attr_2 | attr_3 | ... | attr_n |
| --- | --- | --- | --- | --- | --- | --- |
| XX | XX | XX| XX | XX | ... | XX |
```