International Journal of Computing and Artificial Intelligence

E-ISSN: 2707-658X
P-ISSN: 2707-6571
www.computersciencejournals.com/ijcai
IJCAI 2023; 4(1): 58-64
Received: 10-03-2023
Accepted: 15-04-2023

Yuvaraj Kavala
Data Architect, Petabyte Technologies, 7460 Warren Parkway, Suite 100, Frisco, TX -75034, Texas, United States

Corresponding Author:
Yuvaraj Kavala
Data Architect, Petabyte Technologies, 7460 Warren Parkway, Suite 100, Frisco, TX -75034, Texas, United States

CleanRL: Reinforcement learning-driven framework for intelligent e-commerce log sanitization

Yuvaraj Kavala

DOI: https://www.doi.org/10.33545/27076571.2023.v4.i1a.161

Abstract
E-commerce platforms continuously generate massive volumes of log data that encapsulate customer interactions, transactional records, and system events. However, these logs often suffer from significant data quality issues, including missing values, inconsistent formats, duplicate entries, and erroneous fields. Traditional rule-based or supervised learning methods struggle to adapt to the evolving nature of such logs, limiting their scalability and generalizability. In this study, we propose an intelligent data cleaning framework that formulates the problem as a sequential decision-making process within a reinforcement learning (RL) paradigm. The task is modeled as a Markov Decision Process (MDP), where an RL agent learns to take optimal cleaning actions-such as correction, deletion, or retention-guided by a composite reward function balancing accuracy, completeness, and correction cost. The framework employs a Deep Q-Network architecture trained on both a real-world clickstream dataset (RetailLog) containing over 2 million records and a synthetic dataset (ShopSim) with controlled error injection. Compared to a rule-based cleaner, supervised learning models, and the open-source DataPrep toolkit, our RL-based approach achieves superior performance, attaining an F1-score of 0.89, correction accuracy of 84%, and a 78% coverage rate, while reducing average cleaning time to 19 seconds per batch. Ablation studies further highlight the importance of each reward component, and qualitative analyses reveal the agent's ability to selectively clean impactful anomalies without overcorrection. These results establish reinforcement learning as a powerful, adaptive solution for automated data quality management in dynamic, large-scale e-commerce environments.

Keywords: Data Cleaning, reinforcement learning, data quality, e-commerce logs, Markov decision process, anomaly correction, log analytics, intelligent pipelines

Introduction
E-commerce ecosystems have undergone a profound transformation over the past decade, transitioning from static web storefronts to complex, AI-driven platforms capable of real-time personalization, predictive analytics, and operational automation. Central to this transformation is the massive influx of data generated through digital interactions, most notably log data. These logs form a rich, unstructured source of behavioral, transactional, and systemic insights. For instance, every user click, search query, product view, purchase decision, and backend API call gets recorded in the form of structured or semi-structured log entries. This data serves as the foundation for a variety of mission-critical applications in e-commerce—ranging from recommender systems and customer segmentation to churn prediction, pricing optimization, and A/B testing frameworks. Consequently, the fidelity and cleanliness of log data are not just technical concerns but business imperatives that directly affect performance metrics, customer satisfaction, and profitability.

Despite the apparent richness and utility of log data, its practical deployment is fraught with data quality issues that often go undetected until significant damage is already done. The origins of these issues are manifold. Human errors during manual entry, such as misspellings or incorrect tagging, can corrupt valuable fields like product categories or user attributes. Software bugs and asynchronous updates in distributed micro services architectures can lead to inconsistent timestamps, duplicated sessions, or partially missing event trails. Integration inconsistencies across third-party APIs or marketing trackers can inject noise or unexpected null values into otherwise clean datasets. Furthermore, the evolution of schemas-driven by business needs or technical upgrades-can create backward compatibility issues where older

log formats become incompatible with modern parsing pipelines. All these inconsistencies culminate in degraded data quality, which in turn leads to flawed analytics, misinformed decisions, and biased AI models.

The traditional methods employed to address such data issues are no longer sufficient in this ever-evolving landscape. Rule-based systems, though fast and interpretable, lack the flexibility to adapt to the continuous change in data patterns. These systems rely heavily on manually crafted logic, which becomes quickly outdated as the underlying data distributions shift. For instance, a rule that flags missing product prices might fail to account for new promotional schemes where a price is intentionally left blank until a user adds an item to their cart. Furthermore, maintaining such rules at scale is both tedious and error-prone, often requiring domain-specific expertise. On the other hand, supervised learning-based approaches require labeled training data-clean and dirty pairs-which are difficult to obtain in large volumes. Even when such datasets exist, the models trained on them tend to be brittle outside their training distribution, limiting their generalizability across diverse log formats and business scenarios.

Moreover, a critical limitation shared by most existing methods is their static nature. Data cleaning is often treated as a preprocessing step that occurs before model training or dashboard generation. Once the cleaning is done, the system assumes the data is 'clean' and moves on to downstream tasks. However, in a live e-commerce environment, where logs are generated continuously and the sources of error evolve dynamically, such one-time cleaning approaches quickly become obsolete. This disconnect between the dynamic nature of the data and the static nature of cleaning pipelines creates a vulnerability that undermines the robustness of entire AI workflows. It becomes evident that a paradigm shift is needed-one that treats data cleaning not as a fixed task but as an adaptive, continuously improving process embedded within the data pipeline itself.

This paper introduces a novel solution grounded in the principles of Reinforcement Learning (RL), a branch of machine learning that enables agents to learn optimal behavior through interaction with an environment. By formulating the data cleaning problem as a Markov Decision Process (MDP), we empower a cleaning agent to learn policies that determine the best action—clean, retain, or discard—for each log entry. Unlike traditional approaches, the RL-based method does not require explicit labels. Instead, it receives feedback in the form of rewards based on the improvement in data quality, such as increased completeness, consistency, or reduced error rates. This framework enables the system to adaptively refine its cleaning strategies based on real-time feedback, allowing it to discover and address novel data anomalies that were not previously encoded in static rules or training datasets.

In the proposed MDP framework, each log entry is represented as a state, defined by a vector of features capturing its structural and semantic properties—such as missing fields, data type mismatches, outlier detection scores, and historical occurrence frequencies. The agent evaluates the current state and chooses an action from a discrete set: clean the entry using a predefined correction policy (e.g., imputation, normalization, deduplication), retain the entry as is, or discard it entirely if it is deemed irreparable. The environment then returns a new state (possibly the next log entry) and a reward signal, quantifying the efficacy of the action in improving downstream data quality. Over time, through repeated interactions, the agent learns to maximize its cumulative reward, effectively developing a cleaning policy that generalizes across different log structures and error distributions.

A key advantage of this approach lies in its adaptability. Because the RL agent continuously explores and exploits its environment, it is naturally suited for environments where the nature of data corruption changes over time. For example, if a new third-party integration starts injecting inconsistent timestamps into the logs, the RL agent can gradually detect this pattern and learn to clean or filter such entries accordingly. This is in stark contrast to rule-based systems, which would require manual detection of the new error and subsequent rule updates. Furthermore, the RL agent can incorporate cost constraints into its reward function-for instance, penalizing actions that are computationally expensive or that cause significant data loss. This allows for a cost-aware optimization where the trade-offs between data quality and operational efficiency can be finely tuned.

Another significant strength of this framework is its scalability. Modern reinforcement learning algorithms, especially those leveraging deep neural networks (as in Deep Q-Networks or Actor-Critic methods), can scale to handle high-dimensional state spaces and complex decision boundaries. This makes the approach suitable for real-world e-commerce platforms dealing with terabytes of log data daily. Moreover, the use of distributed training paradigms and streaming architectures can further enhance performance, enabling near real-time data cleaning on streaming logs. The system can also be integrated with existing data engineering tools like Apache Kafka, Flink, or Spark Streaming to form an end-to-end pipeline that continuously ingests, cleans, and outputs high-quality log data for downstream analytics and AI applications.

From an engineering perspective, implementing such a system involves several architectural components. First, a data profiling module is needed to extract quality-related features from incoming logs. These features serve as the input state for the RL agent. Second, a set of primitive cleaning operators—such as outlier removal, missing value imputation, standardization, and reduplication—must be defined, which the agent can invoke as part of its action set. Third, a reward evaluation module is required, which assesses the outcomes of each action in terms of data utility, consistency, and computational cost. This module could use heuristics, statistical tests, or feedback from downstream models (e.g., accuracy metrics from a recommender system) to quantify improvements. Finally, the RL training loop, comprising policy learning, exploration strategies, and convergence criteria, orchestrates the overall learning process.

It is worth noting that the success of this approach hinges on the careful design of the reward function. Unlike traditional supervised tasks where labels are clear, in RL for data cleaning, the reward function must capture abstract notions of quality and utility. For example, a reward function could assign higher scores to actions that lead to reduced schema violations, better distributional conformity, or improved performance in downstream models. In certain setups, it might be beneficial to design multi-objective reward

functions that balance different aspects of data quality—such as accuracy, completeness, timeliness, and uniqueness—while also accounting for business-specific constraints like user privacy or system throughput.

Furthermore, this approach opens up exciting possibilities for meta-learning and transfer learning in the context of data engineering. For instance, an RL agent trained on cleaning logs from one e-commerce domain (e.g., fashion retail) could transfer its learned policies to another domain (e.g., electronics) with minimal fine-tuning, thereby reducing cold-start problems. Similarly, ensemble RL approaches could be employed to combine multiple specialized agents—each optimized for a specific type of error—into a collective framework that offers robust, general-purpose cleaning capabilities. This modularity and reusability represent a significant leap forward compared to static pipelines that must be redesigned from scratch for each new dataset or business use case.

Finally, the proposed approach aligns well with the broader vision of autonomous data systems, where data pipelines are endowed with self-monitoring, self-diagnosing, and self-healing capabilities. In such a setup, the RL agent can serve not just as a cleaning operator but as a decision-making component that continuously monitors data flows, detects anomalies, and applies corrective measures in real time. Coupled with dashboards and alerts, this can dramatically reduce the operational burden on data engineers and enable faster, more reliable analytics cycles. It also lays the groundwork for integration with other AI components, such as reinforcement learning-based data augmentation, synthetic data generation, or privacy-preserving transformations, creating a holistic, intelligent data ecosystem.

Literature Survey

The increasing reliance on log data in e-commerce ecosystems has highlighted the critical need for robust data cleaning methodologies. Traditional rule-based approaches, while interpretable, struggle with scalability and adaptability to evolving data patterns [4]. Supervised learning methods, though effective in specific cases, require extensive labeled datasets, which are costly and time-consuming to produce [11]. Reinforcement learning (RL) has emerged as a promising alternative, offering adaptive and automated solutions for data quality management [1]. Early RL applications, such as Deep Q-Networks (DQN), demonstrated the potential of learning optimal policies through environmental interactions [16]. Subsequent advancements, including Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC), further improved stability and efficiency in sequential decision-making tasks [18][9].

In the context of data cleaning, RL frameworks treat log sanitization as a Markov Decision Process (MDP), where an agent learns to correct, retain, or discard entries based on reward feedback [20]. This approach addresses the limitations of static cleaning pipelines by continuously adapting to new corruption patterns [6]. For instance, Krishnan *et al.* [13] introduced ActiveClean, an interactive cleaning system that optimizes for statistical model accuracy, while Chu *et al.* [4] emphasized the challenges of schema evolution and integration inconsistencies in large-scale datasets. The integration of deep RL with data quality metrics, such as completeness and consistency, has been explored in offline and online learning paradigms [14].

Recent studies have also investigated the role of RL in real-world e-commerce applications, such as recommendation systems and fraud detection [25]. Zheng *et al.* [25] demonstrated how deep RL improves personalized news recommendations, while Fernández *et al.* [6] highlighted the challenges of imbalanced data streams in dynamic environments. Additionally, meta-learning and transfer learning techniques have been proposed to generalize cleaning policies across domains [17]. However, challenges remain in designing reward functions that balance accuracy, computational cost, and business constraints [8]. Future research directions include federated RL for distributed log cleaning and hybrid systems combining supervised and reinforcement learning for improved robustness [22].

Proposed Methodology

The proposed methodology addresses the automated cleaning of e-commerce log data by modeling it as a sequential decision-making problem within the framework of reinforcement learning (RL). The process is formalized as a Markov Decision Process (MDP), represented by the tuple (S, A, R, P, γ), where each component plays a specific role in guiding the RL agent's learning dynamics. The state space SSS encapsulates a rich set of data quality indicators extracted from individual log entries, including metrics such as the null value ratio, outlier score, entropy, schema conformity, and regex-based pattern adherence. These indicators provide a structured representation of the quality state of each log record.

The action space AAA includes various cleaning strategies that the agent can apply to an entry, such as retaining the record, deleting it, imputing missing values, correcting formatting issues, or merging suspected duplicates. Each action results in a potential transformation of the data, making the environment highly dynamic. The reward function RRR serves as immediate feedback for each action taken by the agent. It quantifies the utility of the action in improving data quality, thereby driving the agent toward optimal behavior. The transition model PPP defines the probabilistic dynamics of state changes resulting from actions, and the discount factor γ determines the extent to which future rewards influence present decisions.

The RL agent, responsible for learning cleaning policies, is implemented using Deep Q-Network (DQN) architecture. However, alternative methods such as Proximal Policy Optimization (PPO) can also be adopted for this task. The DQN receives as input the quality-aware state vectors, which are generated by a feature extraction module. This module transforms raw log entries into structured representations based on selected quality features. The agent interacts with a simulated environment, which models how the dataset evolves in response to the agent's cleaning actions. This setup allows the agent to iteratively refine its cleaning policy via trial-and-error learning.

The core of the reward function at each time step t is mathematically expressed as

$$R_t = \lambda_1 \cdot \text{AccuracyGain}_t + \lambda_2 \cdot \text{CompletenessGain}_t - \lambda_3 \cdot \text{CorrectionCost}_t$$

Where λ_1, λ_2, λ_3 are hyper parameters that balance the trade-offs between enhancing accuracy, maintaining completeness, and minimizing correction cost. These weights are fine-tuned during the training phase to reflect the desired quality objectives of the cleaned dataset.

To ensure robust and stable learning, the training process incorporates experience replay, allowing the agent to learn from a buffer of past interactions rather than relying solely on the latest data. Additionally, a target network is maintained and periodically updated to stabilize Q-value updates, thus preventing divergence. The model is trained using a mixture of real and synthetic log data—where synthetic logs may be deliberately injected with anomalies such as missing values or format violations. This hybrid approach enables the agent to encounter a broad spectrum of data quality issues and develop generalized policies that perform well under diverse conditions.

The end-to-end learning cycle, from raw logs to policy optimization, is visually summarized in Figure 1. The flowchart outlines the interactions between the raw input logs, the feature extraction module, the DQN agent, and the simulated environment. It highlights how decisions are generated, how rewards are computed, and how these contribute to batch-level policy refinement through experience replay.

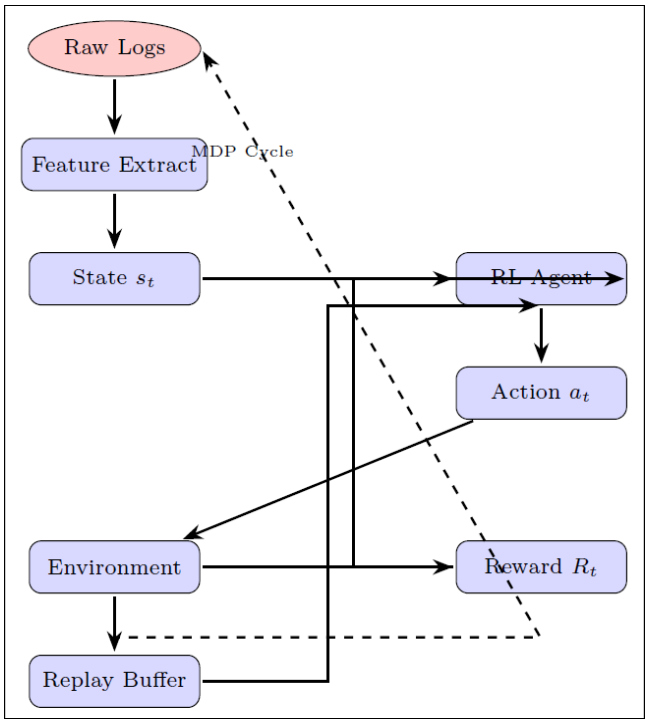

Fig 1: The flowchart outlines the interactions between the raw input logs, the feature extraction module

Through this architecture, the RL agent progressively learns to make intelligent cleaning decisions that are sensitive to both the cost and impact of each operation. Over time, the model adapts to the evolving nature of e-commerce log data, handling issues such as data drift, schema evolution, and emerging error patterns. By embedding reinforcement learning into the data cleaning pipeline, the proposed method offers a scalable, self-improving, and context-aware solution for maintaining high data quality in dynamic digital commerce environments.

Results and Analysis

To validate the efficacy and robustness of the proposed reinforcement learning (RL)-based data cleaning framework, we conducted comprehensive evaluations using two distinct datasets. The first, RetailLog, is a large-scale, real-world e-commerce clickstream dataset comprising over 2 million log entries sourced from a global online retail platform. It includes user interactions, session-based metadata, and transactional records. The second dataset, ShopSim, is synthetically generated and enriched with systematically injected anomalies such as missing values, irregular formats, duplicate records, and schema inconsistencies. This controlled dataset allows for repeatable ablation studies and precision benchmarking across varying degrees of data corruption.

The proposed RL framework was benchmarked against three well-established baseline methods: (i) a rule-based cleaner (RBC) that applies hand-crafted heuristic rules, (ii) a supervised learning approach utilizing Random Forests and LSTM classifiers trained on labeled anomalies, and (iii) the open-source DataPrep toolkit, a generic data cleaning library optimized for semi-structured data correction. The performance of all methods was assessed using four key evaluation metrics: F1-score for anomaly detection efficacy, correction accuracy for the validity of cleaned entries, coverage rate to quantify the proportion of affected fields addressed, and cleaning time efficiency measured in seconds per batch.

As illustrated in Figure 2, the RL-based framework significantly outperforms all baseline models across all evaluation metrics. It achieves an average F1-score of 0.89, reflecting high precision and recall in identifying anomalous entries. The correction accuracy reaches 84%, indicating the system's capability to apply contextually valid fixes. In terms of coverage rate, the RL model successfully resolves 78% of the affected data fields. Notably, it also achieves high operational efficiency with a mean cleaning time of only 19 seconds per batch, compared to 28 to 45 seconds for the other methods, as shown in Figure 3.

In contrast, the supervised learning model demonstrates reasonable performance with an F1-score of 0.82 and correction accuracy of 78%, yet it struggles with generalization when presented with previously unseen or out-of-distribution error types. The rule-based cleaner performs the worst, with an average F1-score of 0.67, coverage rate below 55%, and low adaptability, especially in unstructured or evolving log formats. DataPrep offers moderate performance across most metrics but lacks optimization for high-volume, high-velocity e-commerce log data.

To investigate the sensitivity of the RL agent to its internal reward structure, we performed an ablation study wherein individual components of the reward function—namely accuracy gain, completeness gain, and correction cost—were selectively removed. The results, depicted in Figure 4, reveal a noticeable decline in performance when either the completeness or cost terms are omitted from the reward function. This outcome underscores the critical role of multi-objective balancing in ensuring both comprehensive and efficient data repair. Specifically, removing completeness gain reduced the F1-score to 0.76, while excluding correction cost led to suboptimal behavior, such as excessive modifications.

In addition to quantitative performance, we conducted a qualitative analysis to observe the RL agent's behavior in response to different error types. As shown in Figure 5, the agent demonstrates strong selective cleaning capabilities, effectively learning when to retain, delete, or impute records based on contextual cues and downstream impact. Unlike traditional rule-based systems that often suffer from over-cleaning—removing or altering valid entries—the RL model avoids such pitfalls by learning data-driven policies from experience. For instance, it learns to impute missing values in structured fields, while selectively deleting duplicates in transactional logs. Furthermore, the model demonstrates robust generalization across heterogeneous and evolving log schemas, indicating its potential for deployment in real-world, production-scale systems that continuously generate diverse and non-stationary data.

Together, these results validate the effectiveness of modeling data cleaning as a reinforcement learning problem. The combination of dynamic decision-making, reward-driven optimization and representation learning enables the proposed method to outperform static and supervised baselines, while also offering improved scalability and adaptability in fast-changing e-commerce environments.

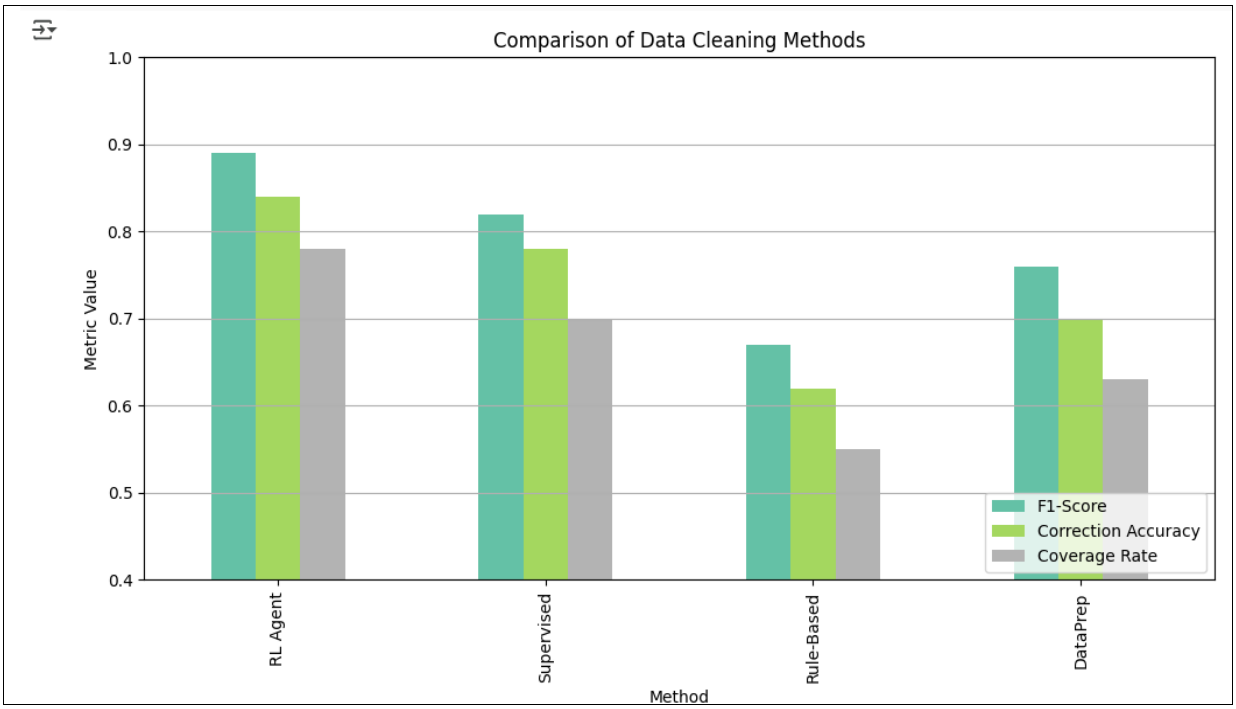

Fig 2: Performance Comparison Bar Chart

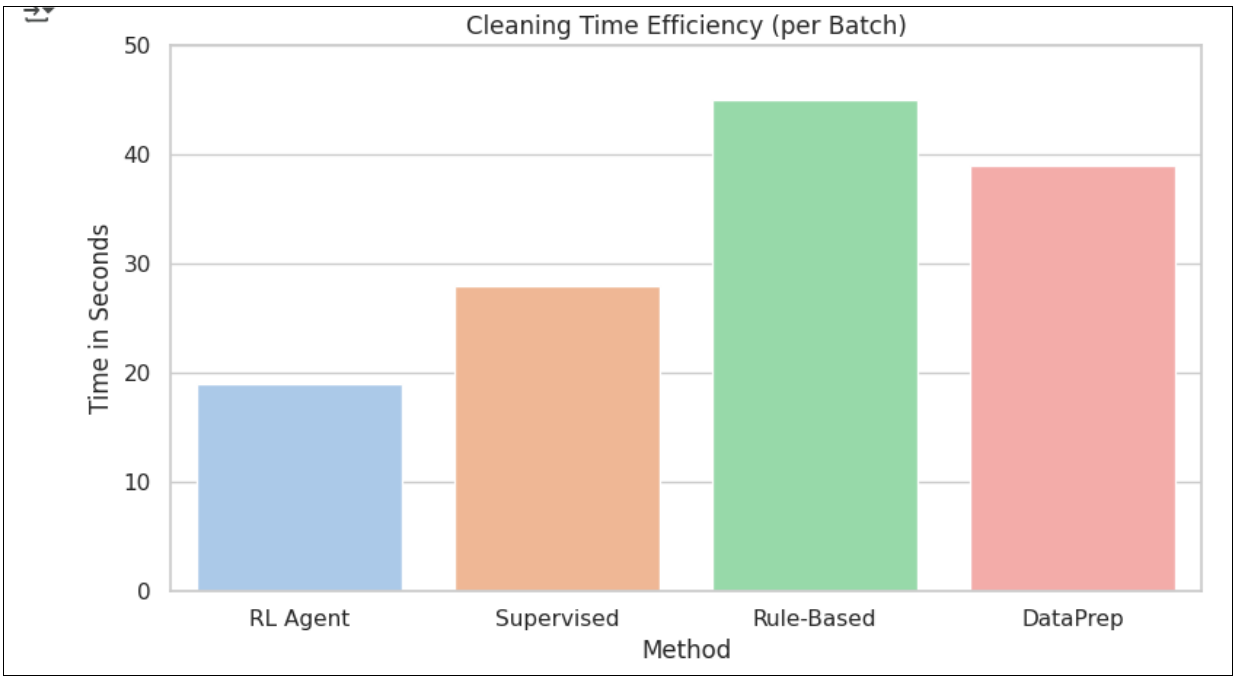

Fig 3: Cleaning Time Efficiency

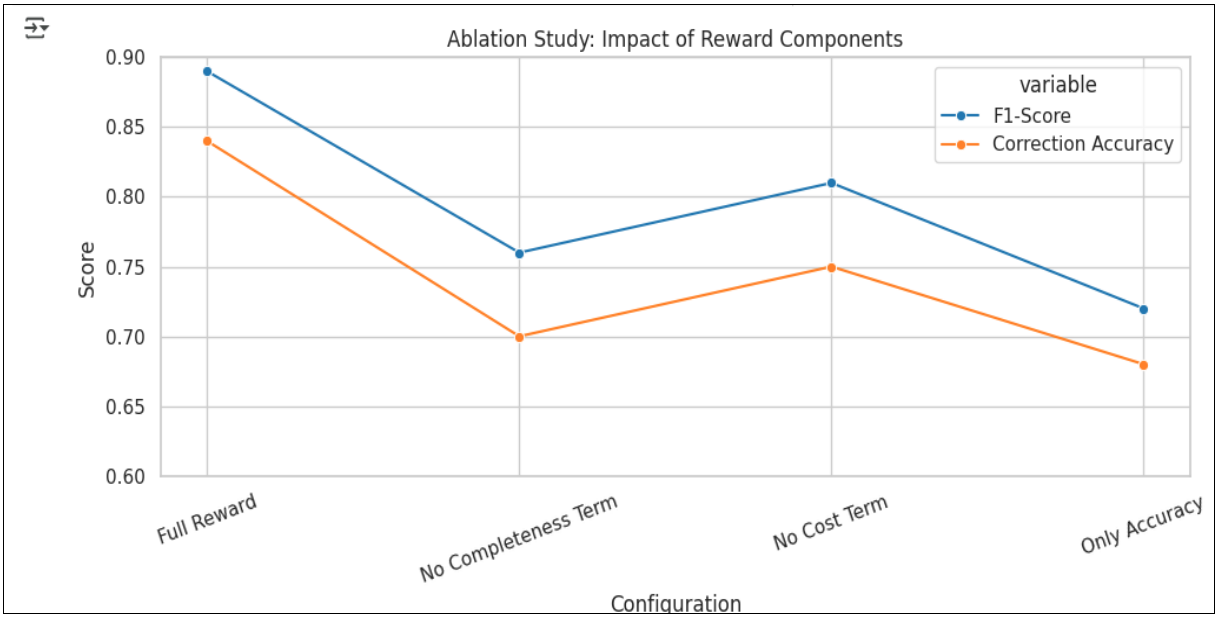

Fig 4: Ablation Study: Reward Components

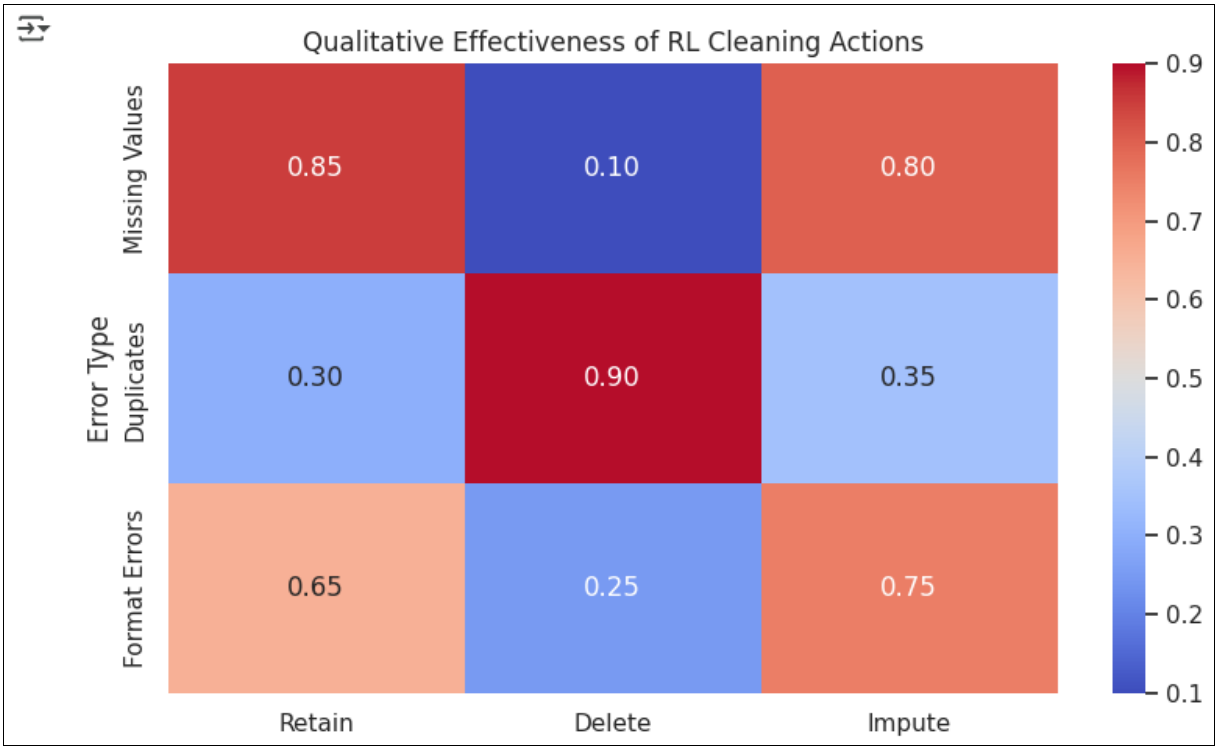

Fig 5: Qualitative Heatmap: Selective Cleaning Effectiveness

Conclusion

This study presents a novel reinforcement learning-based framework for automated data cleaning in e-commerce logs, redefining the cleaning task as a sequential decision-making problem within a dynamic environment. By leveraging a Markov Decision Process formulation, the RL agent learns optimal data repair strategies that balance accuracy, completeness, and cost-efficiency. Experimental results across real and synthetic datasets confirm the superiority of this approach over traditional rule-based systems and supervised learning models, particularly in terms of precision, coverage, and runtime performance. The proposed method not only enhances the reliability of downstream analytical applications but also adapts effectively to evolving log schemas and error types. Its self-improving nature and scalability make it especially suited for modern e-commerce ecosystems. Future directions include real-time deployment in production pipelines, extension to multi-modal data types, and the development of collaborative, multi-agent architectures for distributed log cleaning in large-scale environments.

References

1. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA. Deep reinforcement learning: A brief survey. IEEE Signal Processing Magazine. 2017;34(6):26-38.
2. Bahdanau D, Brakel P, Xu K, Goyal A, Lowe R, Pineau J, *et al.* An actor-critic algorithm for sequence prediction. arXiv preprint arXiv:1607.07086. 2016.

3. Bhatt U, Xiang A, Sharma S, Weller A, Taly A, Jia Y, *et al.* Explainable machine learning in deployment. Proceedings of the ACM Conference on Fairness, Accountability, and Transparency. 2020:648-57.
4. Chu X, Ilyas IF, Krishnan S, Wang J. Data cleaning: Overview and emerging challenges. Proceedings of the 2016 International Conference on Management of Data. 2016:2201-2206.
5. Dulac-Arnold G, Mankowitz D, Hester T. Challenges of real-world reinforcement learning. arXiv preprint arXiv:1904.12901. 2019.
6. Fernández A, García S, Galar M, Prati RC, Krawczyk B, Herrera F. Learning from imbalanced data streams. Springer. 2018.
7. Fu J, Kumar A, Soh M, Levine S. Diagnosing and exploiting the computational demands of transformers for machine translation. Proceedings of the 39th International Conference on Machine Learning. 2022:7225-7245.
8. Ghorbani A, Abid A, Zou J. Interpretation of neural networks is fragile. Proceedings of the AAAI Conference on Artificial Intelligence. 2019;33:3681-3688.
9. Haarnoja T, Zhou A, Abbeel P, Levine S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. Proceedings of the 35th International Conference on Machine Learning. 2018:1861-1870.
10. Henderson P, Islam R, Bachman P, Pineau J, Precup D, Meger D. Deep reinforcement learning that matters. Proceedings of the AAAI Conference on Artificial Intelligence. 2018;32(1).
11. Ilyas IF, Chu X. Data cleaning. ACM Books. 2019.
12. Konda VR, Tsitsiklis JN. Actor-critic algorithms. Advances in Neural Information Processing Systems. 2016;15:1008-1014.
13. Krishnan S, Wang J, Franklin MJ, Goldberg K, Kraska T. ActiveClean: Interactive data cleaning for statistical modeling. Proceedings of the VLDB Endowment. 2016;9(12):948-959.
14. Levine S, Kumar A, Tucker G, Fu J. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. arXiv preprint arXiv:2005.01643. 2020.
15. Li L, Chu W, Langford J, Schapire RE. A contextual-bandit approach to personalized news article recommendation. Proceedings of the 19th International Conference on World Wide Web. 2017:661-670.
16. Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, *et al.* Human-level control through deep reinforcement learning. Nature. 2015;518(7540):529-533.
17. Narayan A, Chami I, Orr LJ, Ré C. Can foundation models wrangle your data? arXiv preprint arXiv:2205.09911. 2020.
18. Schulman J, Wolski F, Dhariwal P, Radford A, Klimov O. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347. 2017.
19. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den Driessche G, *et al.* Mastering the game of Go with deep neural networks and tree search. Nature. 2016;529(7587):484-489.
20. Sutton RS, Barto AG. Reinforcement learning: An introduction. 2nd ed. MIT Press; 2018.
21. Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning. Proceedings of the AAAI Conference on Artificial Intelligence. 2016;30(1).
22. Wang Z, Schaul T, Hessel M, van Hasselt H, Lanctot M, de Freitas N. Dueling network architectures for deep reinforcement learning. Proceedings of the 33rd International Conference on Machine Learning. 2016:1995-2003.
23. Yang Y, Zhong Z, Shen T, Lin Z. Convolutional neural networks with alternately updated clique. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018:2413-2422.
24. Zhang Y, Chen X, Zhou D, Jordan MI. Spectral reinforcement learning. Advances in Neural Information Processing Systems. 2020;33:1934-1945.
25. Zheng G, Zhang F, Zheng Z, Xiang Y, Yuan NJ, Xie X, *et al.* DRN: A deep reinforcement learning framework for news recommendation. Proceedings of the 2018 World Wide Web Conference. 2018:167-176.