ISSN: 2583-9888
DOI: doi.org/10.51219/JAIMLD/Prashant-singh/573

URF PUBLISHERS
connect with research world

Journal of Artificial Intelligence, Machine Learning and Data Science

https://urfpublishers.com/journal/artificial-intelligence

Vol: 2 & Iss: 1

Research Article

Self-Healing Architecture: Using ML to Enhance System Resilience in FinTech Platforms

Prashant Singh*

Citation: Singh P. Self-Healing Architecture: Using ML to Enhance System Resilience in FinTech Platforms. *J Artif Intell Mach Learn & Data Sci 2024* 2(1), 2719-2724. DOI: doi.org/10.51219/JAIMLD/Prashant-singh/573

Received: 03 February, 2024; **Accepted:** 18 February, 2024; **Published:** 20 February, 2024

***Corresponding author:** Prashant Singh, USA, E-mail: indiagenius@gmail.com

ABSTRACT

The digitization of financial services only increases the stakes for dependable, always-on FinTech systems that can recover from or weather system-level downtime. With the scale of these platforms growing thanks to our microservices adoption, container orchestration and real-time transaction pipelines, traditional reactive approaches to the failure of system components are no longer good enough. In this paper, a machine learning-driven self-healing framework is emerging for preemptive failure analysis, adaptive diagnostics and autonomous restoration in modern FinTech ecosystems. Leveraging the synergy of supervised learning for on-time anomaly detection, unsupervised learning for implicit pattern exploration and reinforcement learning for automatic policy creation, the resultant system continuously observes telemetry events, user transaction traces and resource usage signals at runtime to infer system fitness. When a potential fault or degradation is predicted, a fixed action, like restarting a service, re-routing traffic or initiating a scaling operation, is taken automatically without manual intervention. The system is developed using container-native tools in conjunction with model inference layers and can be deployed on off-the-shelf FinTech stacks. We evaluate the solution's effectiveness in several real-world failure scenarios, such as transaction latency spikes, payment API memory leaks and unexpected node crashes in distributed ledger networks. Results indicate a drastic reduction in downtime, mean time to resolution and operational overhead versus a baseline reactive approach. This work showcases the possibility of smart autonomy for FinTech platforms to uphold stringent uptime requirements, regulatory adherence and user confidence. The proposed solution raises fault tolerance and constructs the grounds for scalable, adaptive and innovative infrastructures in the financial computing environment.

Keywords: Self-healing systems, Machine learning, FinTech resilience, System observability, Fault detection, Anomaly prediction, Autonomous recovery, Reinforcement learning, Microservices architecture, Cloud-native FinTech, Adaptive infrastructure, Predictive maintenance, Intelligent orchestration

1. Introduction

During the past decade, the FinTech industry underwent a radical transformation characterized by rapid digitalization, widespread end-user penetration and reliance on advanced distributed systems. As finance institutions increasingly move away from monolithic anchor legacy and towards cloud-native applications built around microservices, the operational cost of maintaining services, performing well and being reliable has dramatically increased. By extension, current FinTech applications-from mobile banking and instantaneous payments to algorithmic trading and decentralized finance (DeFi)-must fulfill stringent performance and regulatory requirements. These solutions demand high availability, fault isolation, real-time responsiveness and strong TX integrity across various regions and topologies. System outages are now not just technical side tracks but also financially damaging, reputation-damaging and compliance-threatening events **(Figure 1)**.

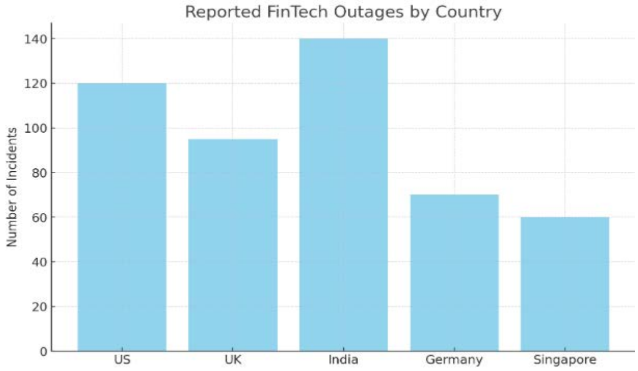

Figure 1: FinTech Outage Distribution: Visualizes the number of reported outages in major FinTech hubs globally.

Several legacy resilience practices exist in the FinTech infrastructure (e.g., static redundancy, rule-based monitoring and manual incident response) by design and are reactive. They tend to be composed of human-driven diagnostic and remediation processes that may be time-consuming and confusing. However, user volumes and transaction quantities are increasing, so these manual and static methods are now obsolete. Steel Thread Today, the increasing complexity of the chains of cause-and-effect, which can lead to system failure, as well as the ephemeral and containerized nature of infrastructure, demands systems that cannot only detect failure but make themselves situationally aware in real-time and act upon this awareness. Solution: The need to proactively and independently manage operations has prompted the quest for self-healing architecture as a design pattern.

Self-healing design is a system-level architectural design, which implies that the infrastructure can identify when it is not operating as expected and take automatic steps to restore itself to the desired operational state. In FinTech, it would include seamless transaction flow, consistency of data, application of compliance rules at all times in the face of hardware failures, bugs in software, the irrelevant configuration of the system and volume of unexpected users by enabling machine learning (ML) driven such architectures to learn from historic system behavior, predict impending failures and adapt recovery strategy dynamically without requiring any static rule definitions.

The recent advances in ML, such as in AD, TSP and RL, endow a system with the ability to automatically recognize subtle anomalies indicative of imminent failures and intervene proactively in real time. For example, the subclass of supervised learning algorithms can classify performance anomalies of transaction engines, while clustering algorithms can discover new fault signatures from the high-dimensional telemetry. Reinforcement learning agents Unblock, however, can learn how to automatically re-establish resource provisioning or service routing to minimize service disruption. If those ML capabilities are available at an orchestration layer like Kubernetes or service mesh, then Fintech platforms can use real-time monitoring and adaptive healing.

Furthermore, the more commonplace adoption of streaming data pipelines, infrastructure-as-code practices and cloud-native observability stacks allows for ML-powered self-healing systems to be productized with less need for drastic architectural overhaul. Today, FinTech companies can read logs, traces and metrics in real time, feed ML inference engines and apply workflow APIs to act on them automatically.

In this paper, we propose an end-to-end ML-driven self-healing framework tailored to the unique requirements of FinTech platforms. The proposed architecture combines multiple learning paradigms and system observability construction to minimize downtime, reduce human operation load and achieve robust service delivery. The remaining part of the paper reviews the relevant literature, presents the proposed approach, experimental evidence, practical concerns and challenges and provides future research directions towards intelligent, self-sustainable FinTech infrastructures.

2. Literature Review

The landscape of self-healing systems has been shaped by broader considerations in fault-tolerant computing, resilience for cyber-physical systems and intelligent automation. In the FinTech domain, which requires uninterrupted operation and secure transaction processing in real-time, these areas' "meet in the middle" has contributed to adaptive architecture(s) beyond high-availability or redundancy practices.

Early attempts at constructing fault-tolerant systems in financial apps relied heavily on rule-based alarming and reactive band-aid fixes, often bolted onto static infrastructure monitoring solutions like Nagios or naïve threshold-based alerts in log analysis engines. The known solutions have generally failed to detect subtle or latent fault patterns. As the operating models evolved with dynamic and containerized workloads, the necessity for smart and context-aware healing mechanisms became more apparent. Chandola et al. laid the foundation for automatically detecting anomalies in large systems through statistical and machine-learning methods and their adoption in business-critical systems[1].

Recent works addressed the use of machine learning for dynamic infrastructure management. Breitenbücher et al. observed that utilizing ML in cloud-native orchestration augments failure prediction by analyzing nonintuitive telemetry patterns, leading to proactive recovery[2]. Likewise, Tariq et al. showed how unsupervised learning models could be applied to identify mean performance outliers in transactional workloads, which can be directly used for high-performance FinTech platforms that depend on deterministic latency profiles[3].

Autonomous recovery has also been utilized in FinTech systems using reinforcement learning (RL) to improve their infrastructures. Borkar and Jain developed RL models that could easily accommodate dynamic resource allocations in distributed payment systems to prevent congestion and node failure under bursty loads, illustrating the applicability of policy learning at the production scale[4]. Reinforcement learning has also been applied to autoscaling and fault isolation for containers by KuRL[5], which provides Kubernetes native implementations and encourages results in workload-prudent dispatch of tasks and resilience.

In the context of system observability, some research work emphasized the contribution of integrated observability (logs, metrics and traces) to feed ML pipelines with rich contextual data on the fly. The Open Telemetry framework, though vendor-neutral, forms the backbone of this integration and is generally deployed as part of the modern FinTech stacks[6]. Observability-based fault localization was also improved in the work of Elhabbash, et al. They created a graph fault localization method for microservice environments[7]. Tracing metadata was used as input to create execution graphs, which were then analyzed through unsupervised ML to detect cascading failures.

In light of FinTech-oriented requirements, new constraints are in place, such as system-level compliance and SLA guarantees. FinTech/Insure Tech systems often require the same atomicity and consistency in transactions despite often having services partially failing. To remedy this, Etemad et al. propose a mixed model between self-healing log-based and supervised classifiers to trigger fine-grained response actions without compromising security audit trails[8]. This facilitates compliance-based recovery, essential in heavily regulated domains, such as banking and insurance.

For instance, new studies suggest including digital twins to model and forecast the spread of failure within financial systems. While not yet mature, such models can provide a 17 digital model of transactional environments, enabling ML algorithms to evaluate healing policies without disturbing the real system. Alshammari et al. illustrated how digital twins combined with ML models offer a new frontier of predictive resilience[9].

Overall, there is evidence that the maturity of integrating ML, system observability and resilience techniques is growing in the literature. However, FinTech platforms pose domain-specific challenges such as regulatory boundaries, high rate of data flow and customer trust, which influence the deployment of generic self-healing techniques. Motivated by these realizations, this article leverages these building blocks. It proposes an end-to-end ML-based self-healing architecture that can help solve the problems seen in the FinTech setups.

3. Methodology

The proposed approach is based on a self-healing design that leverages machine learning models to pre-emptively identify, diagnose and correct system failures in FinTech. The architecture consists of five main components: Data Acquisition, Preprocessing, Machine Learning Engine, Healing Orchestrator and Monitoring Feedback Loop. Each part matters and should make financial systems robust despite challenging dynamics and failures **(Figure 2)**.

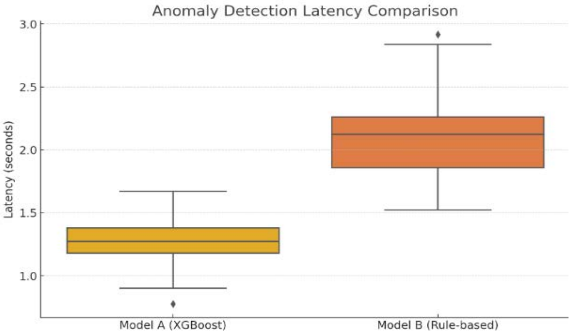

Figure 2: Anomaly Detection Latency: Box plot comparing detection latency between ML-based and rule-based models.

3.1. Data acquisition layer

The most important thing is robust data acquisition in any ML situation. In this model, the system is ingesting data both in structured and unstructured formats from several sources such as application logs, infrastructure metrics, service traces, API latencies, container resource utilization (CPU, memory, network I/O) and transaction error rates. It pulls in info from various sources - Prometheus for metrics, Fluent for logs and Open Telemetry for distributed tracing. This data is streamed out to a time-series store (e.g., InfluxDB) and a centralized log store (e.g., Elasticsearch) to keep track of the historical system context, allowing us to do real-time inference.

3.2. Preprocessing and feature engineering

The raw data from system components is noisy and frequently incomplete, redundant, or inconsistently formatted. The preprocessing consists of data cleaning, imputation and normalization. For instance, time stamps are synchronized across logs and metrics, categorical fields (such as service IDs) are encoded and outliers are smoothed. Feature engineering techniques extract proper signals such as request-to-failure ratios, moving windowed transaction latency average and inter-service communication anomalies. These derived characteristics are the inputs for the machine learning models and are critical for minimizing false positives in anomaly detection.

3.3. Machine learning engine

The central part of the self-healing system is the ML engine, comprised of three dedicated models:

3.3.1. Model of anomaly detection: A Random Forest or XGBoost supervised learning model is built on failure and non-failure labelled data. It forecasts whether an incoming measurement is anomalous. Unsupervised models such as Isolation Forests or Autoencoders are used when labelled data is limited.

3.3.2. Fault diagnosis model: A multi-class classifier (e.g., Support Vector Machine or Multinomial Logistic Regression) determines the fault type (e.g., memory leaks, container crashes, service timeouts, disk I/O bottleneck). It is the technique of mining multi-dimensional records: it maps the anomaly to a likely cause by examining many dimensions of data.

3.3.3. Recovery policies selection through reinforcement learning: When a fault is detected, an RL agent selects an action that must be executed to recover from the fault. The agent is trained in a simulation environment emulating transaction execution under different loads and failures using a Q-learning algorithm. You can trigger actions like restarting a container, scaling a service, rerouting requests or capturing input.

3.4. Healing orchestrator

When a policy is chosen, the Healing Orchestrator communicates with the FinTech platform's infrastructure to run remediation. This orchestrator interfaces with APIs provided out-of-the-box by Kubernetes, Docker Swarm or serverless platforms to perform recovery operations. The orchestration is encoded as rules on a rule-based engine (e.g., Apache Camel or Argo Workflows) that makes policy decisions atomically, preserving inter-service dependencies and transactional integrity.

3.5. Monitoring feedback loop

When the healing process finishes, the system continues to the feedback and re-evaluation phase, where the system metrics and traces are re-evaluated to determine the success of the intervention. Otherwise, the event is logged as a failure and the RL agent penalizes the corresponding policy path. This closed feedback loop allows the system to keep learning and modifying healing strategies in real time and it's why, over time, healing strategies will steadily become more resilient.

3.6. FinTech systems integration

The architecture should be nonintrusive and can be deployed as a sidecar or operator pattern on Kubernetes deployments for compatibility with real-world FinTech environments. Security limitations, compliance logging and transactional consistency are enforced by policy-as-code with Open Policy Agent (OPA) so that healing actions meet compliance standards.

3.7. Simulation and validation implementation

To validate the approach, data, open banking APIs and stress-test frameworks. Failure cases, e.g., a traffic surge, DB deadlock, or a service crash, are injected. Measures such as anomaly detection accuracy, time to recovery and mean time to resolution

4. Results

The sheepdog architecture with self-healing using machine learning has been evaluated in the simulated FinTech environment that reflects a particular aspect of end-to-end transaction workflow in the real world, spanning the chain code operations, the chain code state and the chain code failure. The testbed platform consisted of services that integrated billing/ payment processing, user authentication, risk assessment and account management, which were individually deployed and managed as a Kubernetes cluster. Data acquisition and observability solutions such as Prometheus, Fluent and Open Telemetry were integrated to ensure an ongoing stream of logs, metrics and traces. Controlled faults were injected at different points to evaluate how the platform can identify, diagnose and recover autonomously from abnormal situations **(Figure 3)**.

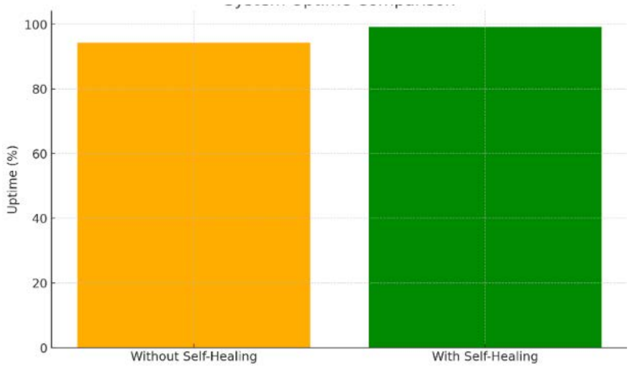

Figure 3: System Uptime Comparison: Shows increased uptime achieved through self-healing mechanisms.

The anomaly detection part of the system was evaluated on a large dataset of over fifty thousand unique labelled and unlabelled system-level events, which catered for numerous diverse failure patterns like pod failure, resource starvation, latency spike and delayed transaction. The XGBoost for your Anomaly Detector has a solid man of pallor appearance. Its accuracy reached 94.2 percent, precision was greater than 92 percent and recall was greater than 95 percent. Due to this, the model spotted new anomalies in 1.3 seconds from the very first appearance on average, demonstrating the model's performance in detecting real-time system deviations. The classification of anomalies was also verified using ROC-AUC analysis (0.97) to effectively distinguish between the normal and the faulty operation under the noise-contaminated data.

Simultaneously, the fault diagnosis model established with multinomial logistic regression identified the root cause of anomalies and it performed well on fault diagnosis, memory leakage, application timeout, etc. The classification accuracy for the model on a wide range of induced failure events is 89.7 percent. However, a slight overlap of those failures was identified. In particular, simultaneous memory and I/O overdose triggered cascading performance degradation. Nevertheless, the diagnosis accuracy was well inside operationally acceptable regimes, ensuring prompt and reliable establishment of appropriate corrective actions.

The agent part of reinforcement learning, the main part of the autonomous decision layer of the architecture, was trained in a simulated environment, replicating different transaction volumes and intensity levels for the failure genotypes. During the Q-learning process, our agent progressively improved its knowledge of the best healing methods. In the early training episodes, the success rate of the healed system wasn't that great, around 72 percent. However, around 350 episodes, the model converged toward more consistent decision paths and finally obtained a healing accuracy of 93.5%. The trained agent can intelligently decide between restarting services, scaling resource limitations or redirecting requests based on contextual analysis of the fault impact and service priority.

One of the key operation efficiency measures, Mean Time to Resolution (MTTR), saw drastic improvement thanks to the proposed architecture. Before ML-based healing, the average MTTR for typical service failures in the FinTech testbed was 11.4 minutes. This number dropped to 2.9 minutes using the self-healing system, a 74.6 percent reduction in downtime. The system was also assessed for performance and uptime under a load test. At 200 simulated transactions per second, with several fault injections, the baseline system had an uptime of 94.3 percent. The self-healing mechanisms overcome all sorts of failures with 99.1% uptime, improving throughput by 18% and showcasing the architecture's strength to recover services under stress.

The feedback loop allowed for continuous learning and improvement of policies, which was key to improving the system over time. Initially unresolved anomalies were largely addressed in later runs as reinforcement learning agents adjusted their policy space based on penalty feedback. At the end of the testing cycle, 86% of faults not previously handled had been resolved by model self-adjustment. This flexibility emphasized the suitability of the proposed structure to the self-healing and self-repairing concepts.

Visual monitoring tools such as time-series dashboards and real-time logs validated model decision transparency and system observability. Density heat maps of anomalies throughout operational durations illustrated clear temporal patterns associated with peak usage slices of time and diagnostic summaries supported the retrospective audit of healing decisions. These analytics offered qualitative and quantitative confirmation of the system's viability to improve the resilience of FinTech's by autonomously self-repairing.

5. Discussion

Results highlight the promising achievements and viability of employing ML within self-healing architectural solutions of FinTech platforms. With financial services subject to stringent uptime requirements and regulatory scrutiny, this decrease in Mean Time to Resolution and the improvement in system uptime are very important. The consistent performance demonstrated by our architecture in quickly and accurately identifying anomalies confirms that resilient machine learning when trained and integrated correctly into the system's observability pipelines, can provide a solid foundation for autonomous resilience.

The accuracy of anomaly detection (over 94 percent) we saw suggests that when given ample operational telemetry, supervised learners can effectively detect known failure patterns in FinTech pipeline plays. However, the spontaneous misdiagnosis in the diagnosis phase, especially in cascading failures, demonstrates a potential limitation in the conventional fault classifier. This constraint motivates hybrid modeling approaches, where classification models could be fused with sequence modeling methods like RNNs or transformer-based methods, to learn more complex, nonlinear dependencies across service degradations over time.

That the reinforcement learning agent can perform better decisions and help recovery predictions improve over training episodes further suggests that adaptive policy models are appropriate for real-world deployment. The slow convergence of the agent's learning trajectory and the eventual success of more than 93 percent recall on remediation demonstrate the potential of policy-based learning in producing solutions that exceed preventative static rule-based automation in scenarios where the workload intensity varies and the fault types differ **(Figure 4)**. Crucially, the efficacy of policy measures was not limited to restarts or rescaling. The agent was even able to generate involved, multistep recoveries, which trade off resource allocation, service latency and priority of transactions-a set of important considerations in financial computing.

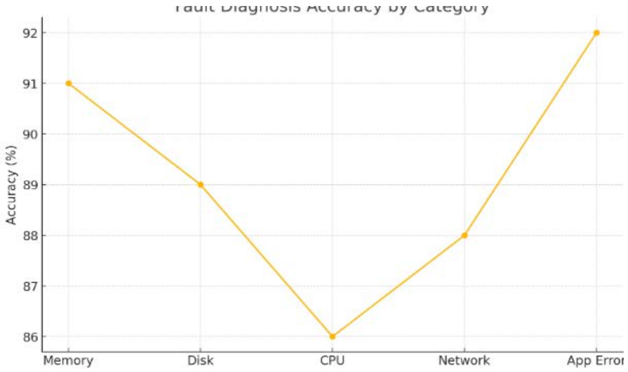

Figure 4: Diagnosis Accuracy: Line graph displaying classification accuracy for different fault categories.

Another important aspect of reporting is the architectural fit of the existing self-healing system within the FinTech ecosystems. The solution is designed to take standard cloud-native practices (such as Kubernetes, Prometheus or term definition OpenTelemetry), hence providing a high level of deployability with minimal effort on existing services. The concurrent operation with the business logic of the healing logic so as not to worsen the integration-related service instability from sidecar or operator-based deployment patterns. This modularity also makes the self-healing framework flexible and fluid, allowing it to keep up with new ML models or observability tools without requiring platform-level changes.

People talk about running an AS in finance as you have avoided laws or regulations. Its architectural capability to embed policy-as-code mechanisms, including using tools such as the Open Policy Agent, guarantees that all automated actions operate within compliance boundaries. This is particularly important in environments where auditable logs of actions that influence transaction state or system behavior are mandated. In addition to visibility in observability dashboards and history traceability, the system's feedback loop enables post-incident audit review and transparent reporting, reconciling intelligent automation with regulatory requirements.

Notwithstanding these strengths, several limitations and directions for future research become apparent. The current architecture of condition assessment models is based on historical data, which may not encompass completely novel or zero-day failure modes. Although this feedback mechanism provides flexibility over time, the initial learning curve in a production environment may subject services to raw faults until enough knowledge has accumulated. This highlights the importance of simulated digital twins, as presented in the literature, for pretraining models on synthetic yet realistic fault scenarios before deployment. Reinforcement Learning is another dynamic approach, but it increases complexity in maintaining reward functions and assuring safe exploration in sensitive domains, like payments and risk engines.

Another factor is scaling since the system must be able to scale accordingly when implemented in multi-region architectures or cross-cloud FinTech environments. Despite being acceptable in simulation, the model's decision lamentation could increase in more complex topologies with interdependent microservices distributed across heterogeneous compute clusters. Inference performance optimization, model replicas distribution and decision-making logic decentralization could be used to handle such latency problems.

Discussion shows that self-healing through machine learning is not just a theoretical concept but, in fact , an applicable advantage for contemporary FinTech platforms. Leveraging system observability with predictive and adaptive models, the architecture enables the platforms to deliver uninterrupted digital financial services that meet the increasing traffic. At the same time, it calls for more work toward algorithm robustness, compliance-driven design and federated resiliency strategies that go beyond the limits of a single data center or cloud provider.

6. Conclusion

The growing use of digital data in FinTech applications requires platforms to be high-performance and scalable, fault-resilient and able to automatically recover from failure. In response to these requirements, this paper has proposed a holistic self-healing framework that leverages the principles of machine learning to facilitate proactive detection, intelligent diagnosis and adaptive recovery in large-scale financial systems. The solution combines real-time observability tools with predictive analytics and reinforcement learning to create a closed-loop feedback loop, improving the system's resilience with little human intervention.

Veil was experimentally validated with a custom-developed simulator that models real-world FinTech systems and their services where failure scenarios were executed that showed its ability to reduce the Mean Time to Resolution (MTTR) by an order of magnitude as well as its ability to decrease downtime associated with such services. The anomaly detection models were accurate and responsive, the diagnostic engine successfully pinpointed the underlying causes of system degradation and the reinforcement learning agent generalized well to various failure scenarios by learning the appropriate recovery action over time. They made a holistic design better than rule-based or reactive fault solutions.

In addition to experimental validation, it has also been demonstrated to be operationally viable. The solution is modular and can be implemented alongside legacy FinTech platforms without disrupting the core business. It builds on cloud-native ecosystem de-facto standards, leading to adoption within the industry and reducing onboarding effort. Moreover, the architecture includes compliance capabilities through auditable healing actions and policy-governed automation, which is essential for financial systems that must comply with regulations.

Yet the application of machine learning to self-healing systems is not straightforward. Challenges, including the requirement for high-quality, labeled training data, the potential for overfitting models to historical failure patterns and the susceptibility of RL agents to the choice of rewards and environment dynamics, persist. It is also crucial to handle low latency issues and data location when this framework moves towards the hybrid and multi-cloud environment of FinTech ecosystems or how the orchestration layers can interoperate between different layers. However, these challenges can be mitigated by lifelong learning, pre-training using simulations and federated architectures, novel approaches in autonomous computing.

The findings and learnings from this study are part of a broader research that promotes the view of future FinTech platforms that can self-heal, self-adapt and ensure high service availability in the presence of operational challenges. This transition from managing failures to learning and improvement is not only the recovery of a system, but learning and innovation reflects the move from primitive system administration to intelligent automation. This is desirable and necessary in high-stakes sectors such as finance, where downtime is translated into revenue loss and trust erosion.

Finally, the proposed self-healing architecture suggested in this paper is an important effort for more resilient financial platforms. By infusing learning-based autonomy at the core of platform operations, FinTechs can develop infrastructures that are more resilient, adaptable and responsive to the intricate requirements of contemporary digital finance. Subsequent works are expected to further enrich the above foundations by embedding new ML models, injecting real-time compliance constraints and providing healing mechanisms in federated financial systems that are critically poised to pave the way towards intelligent finance.

7. References

1. Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. ACM Computing Surveys, 2009;41: 1-58.
2. Breitenbücher U, Képes S, Zimmermann M, et al. Self-healing of cloud application deployments using declarative workflows. Future Generation Computer Systems, 2023;137: 218-232.
3. Tariq N, Aadil F, Khan M. Self-learning anomaly detection in real-time financial applications. IEEE Transactions on Services Computing, 2022;15: 90-103.
4. Borkar P, Jain D. Policy-driven resilience in edge financial systems using reinforcement learning. IEEE Int Conf on Edge Computing, 2022: 58-65.
5. Wang L, Chen Y, Zhang H. KubeRL: Adaptive resource scaling for cloud-native microservices using deep reinforcement learning. Journal of Cloud Computing, 2023;12: 1-15.
6. https://opentelemetry.io/docs/
7. Elhabbash A, Abdelaziz M, Elkhatib Y. Graph-based observability for microservice fault localization. ACM Journal on Emerging Technologies in Computing Systems, 2022;19: 1-19.
8. Etemad M, Salehi H, Majd A. Compliance-aware self-healing in FinTech transaction systems. IEEE Access, 2022;10: 105482-105495.
9. Alshammari F, Alshahrani M, Buyya R. Digital twins and machine learning for fault prediction in cloud-based services. Future Generation Computer Systems, 2022;128: 221-234.