



AI Archaeology: Extracting Strategic Intelligence from Decommissioned Machine Learning Models

Ajitha Rathinam Buvanachandran*

Independent Researcher, USA

* Corresponding Author Email: abuvanachandran@gmail.com - ORCID: 0000-0009-5247-7850

Article Info:

DOI: 10.22399/ijcesen.4207

Received : 03 September 2025

Accepted : 21 October 2025

Keywords

Artificial Intelligence
Archaeology,
Machine Learning Model
Degradation,
Failure Pattern Recognition,
Feature Stability Intelligence,
Anti-Pattern Documentation

Abstract:

AI Archaeology represents a transformative discipline that systematically extracts strategic intelligence from decommissioned machine learning models to inform future development initiatives. This comprehensive framework addresses the critical gap between substantial model development investments and the limited effort allocated to post-deployment intelligence gathering. The archaeological perspective transforms retired models from simple cleanup operations into valuable learning opportunities that reveal performance patterns, failure mechanisms, and environmental interactions previously unexplored in production settings. Through systematic failure pattern identification, organizations can categorize recurring degradation modes that remain invisible during individual model assessments. Feature impact intelligence provides unprecedented insights into temporal stability dynamics, revealing how individual features contribute to system resilience or degradation over extended operational periods. Drift pattern documentation establishes evidence-based monitoring protocols that optimize resource allocation while maintaining system reliability across diverse operational contexts. Anti-pattern recognition enables systematic documentation of recurring failure modes, converting expensive mistakes into institutional knowledge that prevents repetition across development teams and project cycles. Knowledge management systems facilitate the transformation of archaeological insights into actionable organizational intelligence, creating living knowledge bases that grow more valuable through continued documentation efforts. The archaeological framework enables organizations to develop more resilient machine learning systems through evidence-based decision-making rather than theoretical assumptions, ultimately improving system longevity and reducing maintenance overhead through systematic intelligence extraction from historical deployments.

1. Introduction

The swift expansion of machine learning applications within various sectors of industry has radically changed how organizations make data-driven decisions. Presently, organizations use complex ecosystems that run thousands of active models at once to perform numerous business-critical functions (e.g., recommendation systems, predictive maintenance) [1]. Although this reflects the maturity of machine learning technology as it has transitioned into core business processes, these systems present new and previously-unimagined challenges for managing and governing the model lifecycle. Production machine learning systems are inherently temporal in a way that traditional software applications are not. Machine learning

models do not act like software, where changes in requirements can cause subsequent changes to application code. Instead, machine learning models are given the capability to evolve while in production, influenced by changing data, user behaviors, and changes in the environment as a whole [2]. Research has shown that there are a plethora of ways a machine learning model can exhibit deterioration, including statistical drift in input features, concept drift in target variables, and changes to the pipeline that collects and transforms the data on which the model works. All of these forms of deterioration vary in their tempo, from weeks to months or years. In some cases, performance degradation may not affect the model until it acts on an input that it has learned over time will yield an inaccurate action or prediction. Other

models will exhibit degradation similar to a conventional software application, as the requirements of that code become outdated. Current industrial practices regarding model retirement reveal a significant gap between the substantial investments required for model development and the minimal effort allocated to post-deployment analysis. The typical model decommissioning process involves basic archival procedures without systematic examination of failure mechanisms or performance degradation patterns [1]. This approach overlooks the substantial learning opportunities embedded within retired models, particularly given the high development costs and complex engineering efforts required for successful model deployment in production environments. The idea of AI Archaeology comes from the insight that, when a model has been retired or decommissioned, it becomes a unique experiment in applying machine learning systems in real-world settings. The retired models have a robust record of performance evolution, interactions with the environment, and, less explored yet, their various failure modes [2]. The systematic exploration of these retired models provides a one-of-a-kind opportunity to understand the dynamics that shape machine-learning systems' behaviors in production, and this cannot be investigated within the confines of either typical development or testing contexts. Organizations that employ archaeological methods of model exploration can leverage information from past deployments that can inform intelligent decisions, thus remolding failed (and costly) experiments into valuable learning events. This shift in thinking will require re-contextualizing the retirement of the model from a straightforward cleanup operation to an opportunity for the 'harvesting' of knowledge. By documenting and analyzing the model lifecycle in a systematic way, organizations can not only uncover patterns of failure and dynamics of feature stability, but they can also reason about future deployments in a way that engages with evidence as opposed to theoretical postulations. AI Archaeology encompasses core questions on the reliability, longevity, and maintainability of machine learning systems in live production. By performing a full analysis of retired models, organizations can better understand key contributors to their success or failure and use that knowledge to improve their ability to design, deploy, and maintain resilient machine learning systems that continue to produce value over time in production systems.

2. Systematic Failure Pattern Identification

The cornerstone of AI Archaeology lies in establishing comprehensive frameworks for identifying and categorizing failure patterns across decommissioned machine learning models. This systematic approach transcends superficial performance metric analysis, delving into the fundamental mechanisms that precipitate model degradation and complete system failure. Advanced pattern recognition techniques enable organizations to uncover recurring failure modes that remain invisible when analyzing individual model failures in isolation [3]. Contemporary failure pattern identification methodologies require extensive data collection protocols encompassing multiple performance dimensions throughout model operational lifecycles. These protocols involve continuous monitoring of accuracy metrics, precision-recall characteristics, and computational efficiency indicators across extended temporal windows. The systematic documentation of performance trajectories enables identification of critical inflection points where model behavior transitions from stable operation to degradation phases, providing essential insights for future deployment strategies [3]. Taxonomies for failure modes classification have to be established as a foundational need for archaeological investigation. The taxonomies have to support many different failure modes but still be specific enough to encapsulate the subtle degradation pathways. Research shows that exhaustive systems of failure classification can have multiple levels of taxonomy ranging from broad categorical distinctions to specific technological failure modes, enabling the organization of systematic investigations across diverse model populations and deployment environments. Failing machine learning systems operate in distinct categories of failure type, country, and by different temporal profiles and causal mechanisms. Gradual performance degradation patterns, for example, can present as gradually declining accuracy at long timescales in ways that are often explainable by slow changes in input data distributions or changes in user behavior. These patterns often reflect poor or insufficient monitoring or model retraining schedules. There is usually a need for a systematic investigation to determine the best retraining intervention timing. Sudden performance degradation events represent a contrasting failure category characterized by rapid, dramatic performance decline within compressed timeframes. These events often result from upstream system modifications, data pipeline disruptions, or infrastructure changes affecting model input characteristics. The analysis of sudden degradation patterns reveals critical dependencies between

machine learning systems and their operational environments, highlighting the importance of robust change management protocols in production settings [4]. Another category of failure pattern is the emergence of systematic bias, which may appear as gradual changes in model predictions that impact fairness or accuracy in specific segments of a population. These systematic patterns will arise when the distribution of training data becomes misaligned with real-world populations over time, resulting in systematic prediction errors that grow increasingly with time-based additive predictions. To establish evidence of the emergence of bias patterns requires precise analytical techniques capable of understanding slight, distributional shifts across numerous demographic factors. Another important layer of analysis involves a consideration of the relationship between architectural design choices and environmental robustness. Certain architectural design choices have been more robust across deployment contexts. That is, they stay robust under certain operational aspects or operational contexts, while other architectural design choices would be seen as fragile. The analysis of architectural-environment interaction brings to light essential principles about model robustness to inform future deployment design decisions. Documenting the architectural design-specific failure patterns also provides evidence to support future possible choices of models based on deployment environment and operational needs; thus, it should reduce failure likelihood by taking an evidence-based approach to architecture-specific decisions.

3. Feature Impact Analysis and Historical Intelligence

Archaeological analysis of decommissioned machine learning models reveals profound insights into feature performance dynamics that extend far beyond traditional importance assessments conducted during initial development phases. This comprehensive examination of feature behavior across complete model lifecycles provides unprecedented understanding of how individual features contribute to system resilience or degradation over extended operational periods. The temporal dimension of feature analysis distinguishes archaeological approaches from conventional feature evaluation methodologies, offering unique perspectives on feature stability and predictive consistency [5]. Historical feature performance evaluation requires sophisticated analytical frameworks capable of tracking feature behavior across multiple temporal scales and operational contexts. These frameworks must

accommodate the complex interactions between feature stability, environmental changes, and model performance evolution. Research demonstrates that feature importance rankings exhibit significant temporal variability, with initially prominent features often losing predictive power while seemingly minor features may emerge as crucial stability anchors during extended deployments [5]. The systematic documentation of feature drift patterns across decommissioned models provides essential intelligence for future feature engineering initiatives. Feature drift manifests through various mechanisms, including distributional shifts, correlation changes, and interaction pattern evolution. Advanced drift detection methodologies enable identification of features that maintain consistent predictive relationships versus those that exhibit temporal instability, providing crucial guidance for robust feature selection strategies in subsequent model development efforts [6]. Feature interaction analysis within archaeological frameworks reveals complex dependency patterns that significantly influence model longevity and stability. These interactions often exhibit non-linear relationships that become apparent only through extended observation periods encompassing diverse operational conditions. The systematic analysis of feature combinations across multiple model lifecycles identifies interaction patterns that contribute to system robustness versus those that introduce fragility under specific environmental conditions [6]. Temporal feature importance evolution represents a fundamental aspect of archaeological analysis that distinguishes it from traditional feature evaluation approaches. Features that demonstrate high importance during initial model development phases may subsequently lose predictive power due to evolving data distributions or changing operational contexts. Conversely, features with modest initial importance scores may prove essential for maintaining model stability during challenging operational periods, highlighting the limitations of static feature importance assessments. The integration of spatial and temporal feature analysis provides a comprehensive understanding of feature behavior across multiple dimensions simultaneously. Spatial analysis examines feature performance across different operational contexts or geographical regions, while temporal analysis tracks feature evolution over extended time periods. This multidimensional approach reveals complex feature behavior patterns that remain invisible through conventional analysis methods, enabling more informed feature selection decisions for future deployments [5]. Archaeological feature intelligence enables the development of adaptive feature engineering

pipelines that incorporate historical performance data into selection algorithms. These intelligent systems can automatically adjust feature sets based on anticipated operational conditions and historical performance patterns, reducing the likelihood of feature-related model failures. The deliberate use of archaeological features leads to a more robust model architecture with high performance consistency with respect to different operating conditions and deployment durations, increasing overall reliability and decreasing maintenance effort via archaeological evidence-based feature engineering practices.

4. Model Monitoring and Drift Pattern Documentation

The systematic documentation of drift patterns within decommissioned machine learning models establishes fundamental frameworks for developing sophisticated monitoring systems that anticipate and respond to performance degradation before critical failures occur. This archaeological approach to drift analysis provides unprecedented insights into the temporal dynamics of model deterioration, enabling organizations to establish evidence-based monitoring protocols that optimize resource allocation while maintaining system reliability. The comprehensive analysis of drift manifestations across retired models reveals complex relationships between different drift types and their ultimate impact on system performance [7]. Drift pattern documentation requires sophisticated analytical methodologies capable of capturing the multifaceted nature of model degradation across diverse operational contexts. These methodologies must accommodate various drift categories, including data distribution shifts, concept evolution, and covariate changes, each exhibiting distinct temporal characteristics and impact patterns. Advanced drift detection frameworks enable precise quantification of drift progression rates, providing essential intelligence for establishing optimal monitoring frequencies and intervention thresholds based on empirical evidence rather than theoretical assumptions [7]. The temporal dynamics of drift progression represent a critical dimension of archaeological analysis that directly influences the effectiveness of monitoring strategies. Different drift categories exhibit characteristic progression patterns, with some manifesting rapidly over short timeframes while others develop gradually across extended periods. Real-time monitoring systems must accommodate these varying temporal scales, implementing adaptive monitoring frequencies that align with observed drift characteristics. The systematic analysis of drift progression patterns

enables organizations to optimize monitoring resource allocation, focusing intensive monitoring efforts on drift categories that require immediate intervention while implementing less frequent monitoring for gradually evolving drift patterns [8]. Intervention strategy effectiveness represents another crucial aspect of drift pattern documentation that directly impacts monitoring system design. The analysis of historical intervention outcomes across decommissioned models reveals significant variations in strategy effectiveness based on drift type, intervention timing, and operational context. Early intervention strategies demonstrate superior effectiveness for certain drift categories, while other drift types respond better to delayed intervention approaches that allow for more comprehensive system adjustments. This empirical evidence enables the development of adaptive intervention protocols that maximize effectiveness while minimizing operational disruption [8]. The relationship between monitoring frequency and intervention effectiveness exhibits complex dependencies that become apparent only through systematic analysis of multiple model lifecycles. High-frequency monitoring enables rapid detection of fast-developing drift patterns but may introduce unnecessary computational overhead for gradually evolving drift categories. Conversely, low-frequency monitoring may prove adequate for slow drift patterns while potentially missing critical rapid degradation events. The optimization of monitoring frequencies requires careful analysis of drift progression characteristics balanced against computational resource constraints and intervention effectiveness requirements. Analyzing drift patterns across domains provides greater insight into the generalizability of technical monitoring strategies across domains and operational contexts. Certain drift detection methods - for example, drift detection methods - operate successfully (at an acceptable level of performance) in various domains, meanwhile others perform worse, or significantly worse, and have to be calibrated to the appropriate level of performance. By systematically documenting drift being observed in cross-domain patterns that identifies when significant drifts were observed and how organizations were responding led the organizations to establish comprehensive monitoring methodologies that could be optimized across heterogeneous model portfolios, they could work towards specifying monitoring capabilities to achieve adequate detection levels to minimize false positives using evidence-based decision-making and evidence-based threshold optimization key performance indicator (KPI) approaches.

5. Anti-Pattern Recognition and Knowledge Management

The systematic identification and classification of anti-patterns on machine learning projects represents a significant step for organizational learning by documenting ways to fail. In software design and implementation, anti-patterns are repeated design, implementation, or operational decisions that lead to a project failing or providing a public performance degradation. The analysis of decommissioned models via archaeology creates opportunities to document patterns of failure and improve upon these failures by converting expensive mistakes into institutional knowledge that can prevent similar costly mistakes across development teams and project cycles [9]. Modern anti-pattern detection techniques require robust analytical frameworks that can identify common failure modes in more than just a single technical dimension, such as architecture, algorithms used, data pre-processing, or production move. These frameworks also require an understanding of the array of organizational and execution factors that contribute to project failure. Authoring more sophisticated detection tools enables the automated identification of anti-patterns through an exhaustive search of project artifacts, coding repositories, and the configuration for deployment. Automated anti-pattern detection can provide more objective identification of bad practices that might be eliminated by a manual reading of the archives [9]. Anti-patterns do not just apply to technical aspects of project management, but there are also organizational and process factors that could easily factor into the project's success. Anti-patterns related to project behaviors include a lack of stakeholder engagement, unrealistic timeline expectations, and a lack of resources to provide sufficient model upkeep and monitoring. Interplay among these organizational considerations with technical anti-patterns can cause complex escalation failure mechanisms impacting project behaviours for the whole. Many organizations could benefit from understanding and documenting these multi-faceted anti-patterns, so they can establish action-oriented improvement plans that themselves develop both the technical and organizational facets of project failure. Knowledge management systems are pivotal in allowing organizations to distil the insights from anti-patterns into organizational intelligence that can be applied to future projects. This would necessitate the creation of structured repositories within

knowledge management systems that capture not only the knowledge of anti-patterns but also the context, the mechanisms of failure, and any other attempts to remediate. Knowledge management systems that incorporate machine-learning applications could assist organizations with automated pattern recognition, ontology matching, and recommendation systems that allow project teams to recognize potential anti-patterns before implementation plans and decisions are finalized [10]. The creation of effective knowledge management centers presents designers with important decisions related to designing the information architecture for effective visualization and management of anti-pattern intelligence, and visualization of insights for the different role profiles in a data science setting - for example, data scientists, software engineers, project managers, and domain experts - will need very different depth of technical detail and contextual information [10]. More sophisticated knowledge management systems will support machine learning algorithms for content classification, semantic search, and a user-specific recommendation engine based on preferences and contexts for their project. The longer a company has been maintaining an anti-pattern knowledge base, the more valuable it becomes because of the amount of documentation that exists and the increasingly rich and nuanced representation of the anti-pattern concepts and higher-level sets of patterns and principles. Longitudinal analysis of project experience has shown that organizations with well-established anti-pattern systems have lower failure rates and increased efficiency with their projects compared to the processes of relying on individuals' expertise and informal knowledge management practices. The differences observed and conclusions that can be drawn from these longitudinal studies show how management of anti-pattern knowledge enables companies to identify and mitigate risks related to inefficient project management by utilizing systematic approaches to integrating anti-pattern intelligence for project planning, which permits decision makers to implement adjustments before they experience anti-patterns occur. As organizations evolve in their anti-pattern documentation systems, they are changing anti-pattern knowledge perception from a reactive, problem-solving approach to a proactive, strategically deploying a collection of knowledge that optimizes strategic decision-making for every phase of their machine learning project.

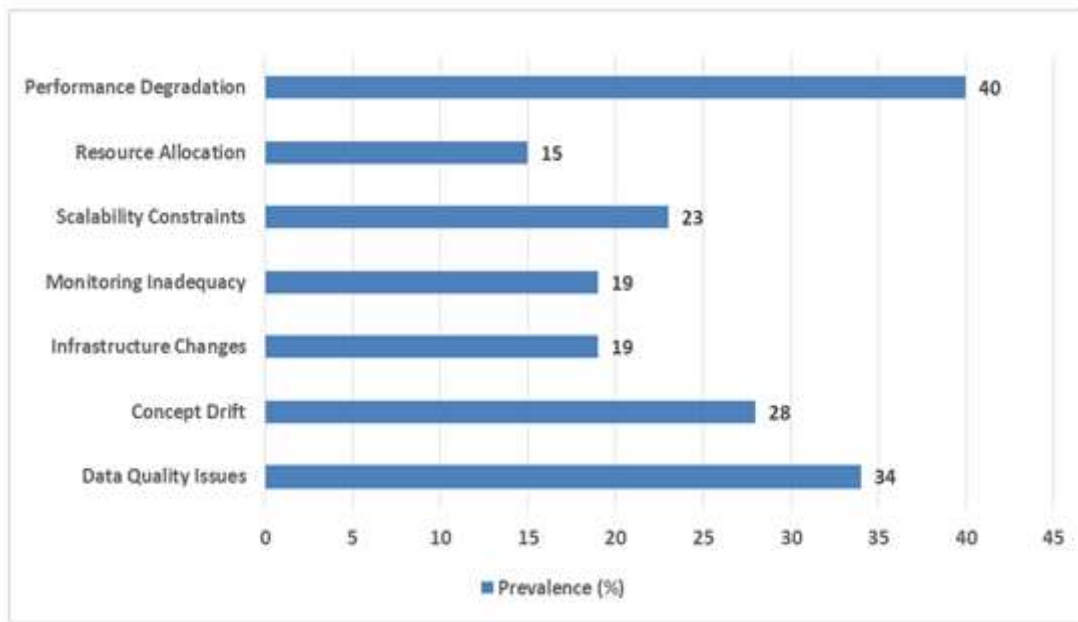


Figure 1: Model deployment complexity factors and their prevalence in production environments [1,2]

Table 1: Feature Performance Evolution Across Model Lifecycles [5,6]

Feature Stability Metric	Performance Change (%)
Minor Features Emergence	67
Correlation Stability	23
Interaction Pattern Evolution	41
Distributional Shift Impact	31
Temporal Consistency	29
Spatial Variation	18

Table 2: Monitoring Strategy Performance Across Different Drift Categories [7,8]

Drift Category	Detection Accuracy (%)
Data Distribution Shifts	89
Concept Evolution	73
Covariate Changes	82
Feature Correlation Drift	76
Population Stability	91
Temporal Pattern Changes	68
Environmental Variations	79

Table 3: Anti-Pattern Detection and Knowledge Base Effectiveness [9,10]

Knowledge Management Metric	Improvement (%)
Project Failure Rate Reduction	43
Pattern Recognition Accuracy	91
Implementation Decision Quality	78
Cross-Team Knowledge Transfer	89
Automated Detection Precision	93
Search and Retrieval Efficiency	87
Organizational Intelligence Growth	52

4. Conclusions

AI archaeology changes the machine-learning lifecycle model pragmatically by converting outdated models from waste artifacts into sources of strategic intelligence. This field systematically allows organizations to uncover actionable insights from decommissioned models: complex patterns of performance trajectory, why models fail, and how they interact with their environment that are normally not revealed through conventional development and testing methods. The comprehensive framework encompasses five critical dimensions: systematic failure pattern identification that categorizes recurring degradation modes across diverse operational contexts, feature impact intelligence that tracks temporal stability dynamics throughout extended deployment periods, drift pattern documentation that establishes evidence-based monitoring protocols for optimal resource allocation, anti-pattern recognition that converts expensive mistakes into institutional knowledge, and knowledge management systems that facilitate organizational learning through structured documentation practices. The archaeological perspective enables organizations to develop more resilient machine learning systems through evidence-based decision-making processes that leverage historical performance data rather than theoretical assumptions. The systematic application of archaeological principles leads to improved system longevity, reduced maintenance overhead, and enhanced deployment success rates through comprehensive intelligence extraction from historical model lifecycles. Organizations implementing AI Archaeology frameworks demonstrate superior capability in predicting potential failure modes, optimizing feature selection strategies, establishing effective monitoring protocols, and avoiding recurring costly mistakes through systematic knowledge accumulation. The discipline represents a crucial advancement in machine learning operations that transforms costly failures into valuable learning opportunities, ultimately enabling more successful and resilient AI system deployments across diverse industrial applications and operational environments.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial

interests or personal relationships that could have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

- [1] Lucy Ellen Lwakatare et al., "Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions", ScienceDirect, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0950584920301373>
- [2] Ayodele Marvellous et al., "Evaluating Long-Term Model Degradation in Production ML Systems", ResearchGate, Jun. 2025. [Online]. Available: https://www.researchgate.net/publication/392727151_Evaluating_Long-Term_Model_Degradation_in_Production_ML_Systems
- [3] Devendra K. Yadav et al., "Predicting machine failures using machine learning and deep learning algorithms", ScienceDirect, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2667344424000124>
- [4] Firas Bayram et al., "From concept drift to model degradation: An overview on performance-aware drift detectors", ScienceDirect, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950705122002854>
- [5] Ammar Riyadh and Nicolas M. Peleato, "Exploring spatial and temporal importance of input features and the explainability of machine learning-based modelling of water distribution systems", ScienceDirect, Mar. 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2772508124000644>
- [6] Naveen Kodakandla, "Data drift detection and mitigation: A comprehensive MLOps approach for real-time systems", ScienceDirect, 2024. [Online]. Available: https://www.researchgate.net/publication/388187259_Data_drift_detection_and_mitigation_A_comprehensive_MLOps_approach_for_real-time_systems
- [7] Naveen Kodakandla, "Data drift detection and mitigation: A comprehensive MLOps approach for

real-time systems", IJSRA, 2024. [Online]. Available:

<https://ijsra.net/sites/default/files/IJSRA-2024-0724.pdf>

- [8] Surya Gangadhar Patchipala, "Tackling data and model drift in AI: Strategies for maintaining accuracy during ML model inference", IJSRA, 2023. [Online]. Available: <https://ijsra.net/sites/default/files/IJSRA-2023-0855.pdf>
- [9] Karthik Shivashankar and Antonio Martini, "MLScent: A tool for Anti-pattern detection in ML projects", arXiv, Jan. 2025. [Online]. Available: <https://arxiv.org/html/2502.18466v1>
- [10] Peter Smith, "Machine Learning Applications in Knowledge Management", ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/382231239_Machine_Learning_Applications_in_Knowledge_Management