International Journal of Computer (IJC)

ISSN 2307-4523 (Print & Online)

https://ijcjournal.org/index.php/InternationalJournalOfComputer/index

Strategies for Managing Technical Debt in the Scaling of Automated Information Processing Systems Using Artificial Intelligence Methods

Dmitrii Bezfamilnyi*

Software Development Expert,Russia / Moscow

Email:dgb991@gmail.com

Abstract

The article presents an analysis of contemporary approaches to managing technical debt in the scaling of automated information processing systems using artificial intelligence methods. The study is conducted in the format of a systematic review and analytical synthesis of scientific publications addressing technical debt in AI systems, scalability issues, and the management of data, models, and architecture. Particular attention is given to the interrelationship between the stages of technical debt formation, system structure, and the quality of decision-making. The main sources of debt accumulation are examined, including errors at the requirements level, data degradation, model instability, increasing architectural complexity, and methodological limitations associated with substituting causal relationships with correlations. It is established that under scaling conditions, technical debt does not lead to system failures but manifests through behavioral distortions and a decline in decision quality. It is substantiated that management effectiveness is determined not by the elimination of individual defects, but by the coordination of actions across all system levels and the ability to constrain the propagation of debt. An original multi-level model of technical debt management is proposed, reflecting the processes of its formation, diagnosis, management, and prevention, while accounting for its impact on system behavior. The model demonstrates a transition from reactive mitigation to proactive architectural management based on the integration of MLOps, data governance, enhanced model transparency, and the use of causal modeling.

Keywords: technical debt; artificial intelligence; scalable systems; data; architecture; causal modeling; system management.

Received: 5/14/2026
Accepted: 7/14/2026
Published: 7/24/2026

----

* *Corresponding author.*

1. Introduction

The rapid development of digital platforms and the transition to artificial intelligence systems have significantly transformed approaches to the design and evolution of software solutions [6]. These systems are no longer isolated analytical tools. They represent complex ecosystems in which data, models, and infrastructure are tightly interconnected. As scaling progresses, both data processing capabilities and latent constraints increase, the latter accumulating over time and manifesting as technical debt.

The relevance of the study is associated with the fact that in AI systems, technical debt accumulates faster and in more complex forms than in traditional systems. It is no longer limited to code, but extends to data, models, infrastructure, and even requirements [9]. In turn, part of the problem originates at early stages due to incorrect or incomplete assumptions about how the system will operate in a real environment [4]. Despite the large body of research, a comprehensive approach to managing such debt remains insufficiently developed. In practice, it is often identified only after problems emerge, which complicates system evolution and increases costs [11]. Therefore, there is a need to consider technical debt as a systemic issue and to manage it across all stages, from design to operation.

The objective of the study is to analyze mechanisms of technical debt formation in scalable AI systems and to develop a conceptual framework for its management across system layers. To achieve this objective, the following tasks are addressed:

– to systematize the types of technical debt in AI systems;

– to identify the main areas of its accumulation during scaling;

– to determine the impact of technical debt on the quality of AI systems;

– to analyze existing strategies for the identification and management of technical debt.

The research hypothesis assumes that technical debt in AI systems emerges as an interconnected multi-layer phenomenon, and that its control is only possible through coordinated system-level architectural and data-driven governance mechanisms.

The scientific novelty of the study lies in reinterpreting technical debt in AI systems not as a set of isolated engineering trade-offs, but as a dynamic and self-reinforcing phenomenon that arises at the intersection of architecture, data, models, and environmental assumptions during system scaling. In contrast to the dominant approaches in the literature, which focus primarily on code and architecture, this work proposes an in-depth analytical perspective in which technical debt is examined through the hidden mechanisms of its generation, including methodological debt (associated with substituting causal relationships with correlations), environmental assumption debt, and system quality degradation at the level of functional correctness. This makes it possible to move beyond the traditional understanding of technical debt and demonstrate that, in scalable AI systems, the primary risks emerge from development processes, decision-making logic, data structures, and interactions with the real-world environment. Unlike existing approaches that primarily address technical debt at the code or architecture level (e.g., MLOps-based monitoring or refactoring strategies), this study extends the concept to include data-induced, assumption-based, and causal-modeling-related debt mechanisms, thereby integrating structural and epistemic dimensions of AI system degradation.

2. Materials and Methods

The study is based on methods of theoretical analysis of scientific publications, categorical classification of technical debt types in AI systems, and comparative analysis of the interrelationships between system architecture, data characteristics, model decisions, and scaling parameters. The main focus is on identifying the mechanisms of technical debt formation in intelligent automated information processing systems, as well as analyzing the factors that determine its impact on system stability, quality, and controllability as complexity increases. Particular emphasis is placed on considering technical debt as a systemic phenomenon arising from the interaction of requirements, data, models, and infrastructure.

The study was conducted in the format of a systematic review of open-access scientific publications for the period 2022–2026, published in international peer-reviewed journals. Source selection was carried out using the databases Google Scholar, ScienceDirect, SpringerLink, and MDPI, based on the following keywords and their combinations: "technical debt in AI systems," "AI technical debt management," "MLOps," "data debt," "model debt," "algorithm debt," "AI system scalability," "AI architecture," "causal machine learning," "requirements technical debt," using logical operators AND/OR. The sample included English-language publications containing empirical or review-based evidence on technical debt in AI systems, issues of scalability, and the management of data, models, and infrastructure. Studies focusing exclusively on narrow development aspects without considering the system context or not related to artificial intelligence methods were excluded.

At the initial stage, 28 publications were selected. After removing duplicates and analyzing titles and abstracts, studies that did not meet the research scope were excluded. Based on the full-text analysis, a final sample of 15 sources was сформed, meeting the stated objectives.

The analytical procedure included sequential stages of source selection, deduplication, thematic filtering, full-text analysis, and categorical systematization of results. During the analysis, the following groups of factors were identified: types of technical debt in AI systems (data, models, architecture, infrastructure, requirements), mechanisms of its formation (assumption errors, data degradation, pipeline complexity, architectural trade-offs), system quality characteristics affected by it, and existing strategies for its identification and management. It was established that the accumulation of technical debt is determined by the degree of interconnection between system components, data quality, model adaptability, architectural consistency, and the presence of monitoring and management mechanisms.

The limitations of the study are related to the thematic scope of the sample. A significant portion of publications in the AI field focuses on individual aspects, such as models or algorithms, whereas issues of systemic technical debt management during scaling remain insufficiently explored. The selected sources cover problems related to architecture, data, models, infrastructure, and AI system quality; however, they do not always consider them within a unified integrated framework.

The obtained results were used to systematize contemporary approaches to technical debt management and to develop an original model reflecting the interrelationships between sources of debt, system characteristics, and strategies for its prevention and mitigation in the context of scaling AI systems.

3. Results

The conducted analysis demonstrates that, as AI systems scale, the nature of technical debt undergoes a fundamental transformation. It ceases to be a localized development constraint and begins to directly influence system behavior. This is the key distinction. In traditional software systems, technical debt slows down change. In AI environments, it distorts decision-making. A system may remain formally operational while producing incorrect outputs. At this point, classical interpretations of technical debt become insufficient [1]. In scalable AI systems, a stable set of technical debt types emerges, defining the limits of their development. These include infrastructure debt, architectural debt, data debt, and code debt. These components are not equivalent and affect the system in different ways; however, together they form an integrated system of constraints [7]. As system load increases, these elements are the first to disrupt controllability and operational stability [13]. Table 1 presents the most significant types of technical debt and their impact on scaling.

Table 1: The most significant types of technical debt in AI systems (Compiled by the author based on source [10])

Type of technical debt	**Frequency**	**Main effect during scaling**	**Strategic priority**
Infrastructure debt	23	Growth of operational constraints and costs	Very high
Code debt	16	Increased complexity of maintenance and modifications	High
Architectural debt	14	Reduced flexibility and integrability	High
Data debt	14	Model degradation and instability of decisions	Critical

The data in the table capture a critical point. Infrastructure debt sets the limit for scalability. As it accumulates, the system does not merely slow down—it loses the ability to operate reliably under load. Failures emerge, dependence on manual interventions increases, and reproducibility deteriorates. As a result, even correctly designed models begin to produce unstable outputs. Thus, the problem extends beyond development and shifts to the level of system operation.

Data debt generates a different type of risk. It does not cause immediate failures; instead, it distorts model behavior. As data volume grows, the number of hidden dependencies and noise increases. The model continues to function, but its outputs become less accurate and less stable [5]. This is particularly dangerous. The error does not manifest explicitly—it is embedded in the decision-making process. In this way, a form of debt emerges that is not captured by traditional control methods.

Architectural debt amplifies both processes. It increases the coupling between components, so that any change begins to affect multiple parts of the system simultaneously. As a result, the cost of change rises, and teams begin to avoid modifications [8]. Temporary solutions emerge and quickly become permanent. This leads to an accelerated cycle of debt accumulation. Further analysis shows that technical debt in AI systems is uneven in terms of criticality. This is reflected in Table 2.

Table 2: Distribution of technical debt categories by severity level (Compiled by the author based on source [15])

Severity level	Number of categories	Nature of impact	Impact on scaling
5 (critical)	5	Systemic, strategic	Directly limits scalability
4 (high)	9	Operational, accumulative	Reduces stability as the system grows
3 (medium)	3	Local	Affects efficiency
2 (low)	1	Limited	Weak impact

Table 2 shows that the majority of technical debt is concentrated at high levels of criticality. This indicates that technical debt in AI systems is inherently systemic in nature. It does not accumulate gradually as a secondary issue; rather, it immediately affects the core parameters of the system. During scaling, this leads to an amplification effect: one type of debt triggers another. Data debt reinforces model debt, while architectural debt complicates infrastructure management. As a result, an interconnected structure of debt emerges that cannot be resolved locally.

Classical approaches suggest addressing technical debt incrementally. In AI systems, this approach proves ineffective. Local fixes to individual components do not stabilize the system [2], as other elements continue to generate new constraints. Consequently, a self-reinforcing cycle of debt accumulation is formed, directly limiting scalability and increasing system complexity.

At a deeper level of analysis, a more profound effect becomes evident. Technical debt impacts not only development and maintenance but also alters system performance quality. Importantly, this influence shifts to the behavioral level. This represents a critical distinction between AI systems and traditional software [14]. Figure 1 presents the quality characteristics most affected by technical debt.

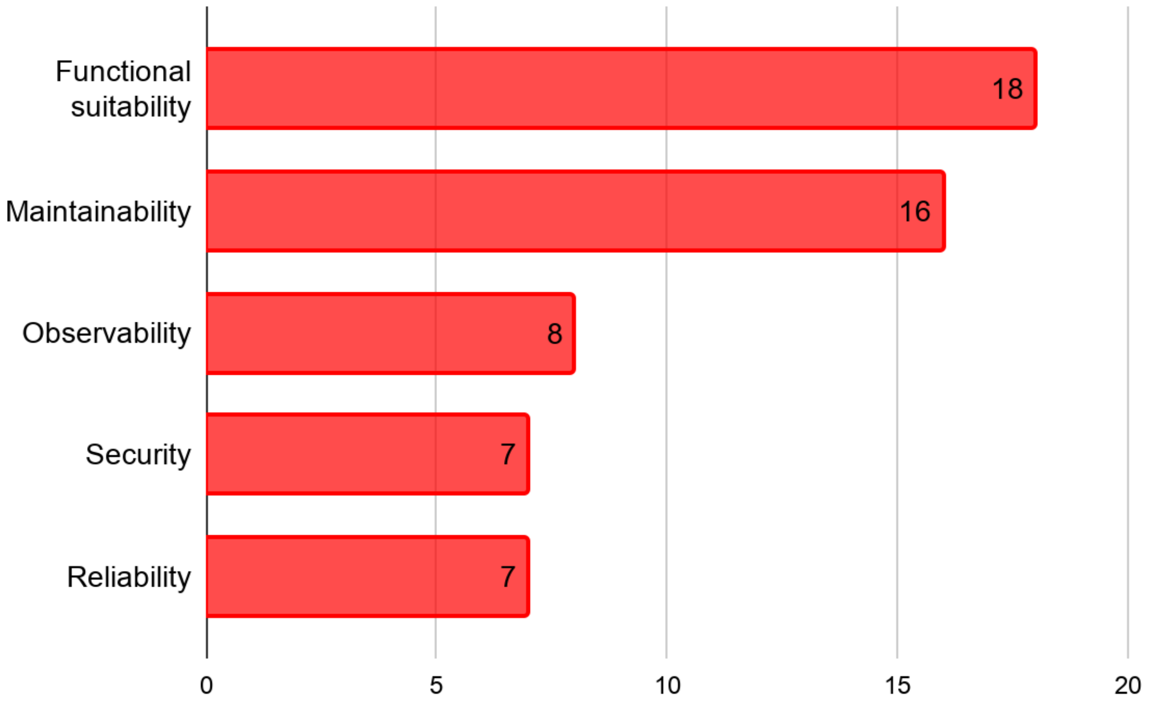

Figure 1: AI system quality most affected by technical debt (Compiled by the author based on source: [3])

The distribution demonstrates that the greatest pressure is exerted on functional suitability, as a result of which the system begins to perform its core functions less effectively. However, failure does not occur. The system continues to operate and generate outputs, but their correctness declines. This situation is particularly dangerous, as the error is not detected as a failure but manifests through a gradual degradation in the quality of decision-making.

Maintainability amplifies this effect. As technical debt increases, the system becomes more complex. Changes require more time, errors accumulate faster, and teams gradually lose control over system logic. As a result, any improvements slow down. However, the primary risk lies elsewhere: the system continues to function despite the deterioration in decision quality.

Observability, security, and reliability further intensify the problem. With insufficient observability, it becomes impossible to accurately identify the source of errors. Reduced reliability leads to unstable system behavior. In conditions of changing data, this results in unpredictable outcomes [12]. Technical debt begins to influence not only the system structure but also the trajectory of its behavior in a real-world environment.

The classical perspective associates technical debt with increasing costs. In AI systems, this is insufficient. Here, technical debt begins to distort the decision-making process itself. A system may remain technically operational while simultaneously generating incorrect results. It is precisely this effect that makes technical debt a key constraint on the scalability of intelligent systems.

In scalable AI systems, technical debt ceases to be a byproduct of development and becomes a factor that determines system behavior under conditions of uncertainty. It defines the boundaries of architecture, infrastructure, and decision correctness. As a result, technical debt evolves from an engineering maintenance issue into a mechanism that shapes the system's development trajectory and effectively determines the limits of its scalability.

4. Discussion

The growth of scalable AI systems is accompanied not merely by an increase in the number of components and data. The very logic of system operation is transformed. Technical debt ceases to be a local engineering issue and begins to form as a latent structure of constraints embedded in architecture, data, and decision-making mechanisms. Under these conditions, managing technical debt requires a reconsideration of fundamental approaches, as traditional methods focused on code and isolated defects are unable to explain system behavior during scaling.

The formation of technical debt begins long before the implementation stage. Errors originate at the level of requirements, where the system relies on implicit or incorrect assumptions about the environment. Such assumptions are not perceived as risks while the system operates under limited conditions. However, as scale expands, they begin to determine system behavior. Requirements no longer correspond to reality, and technical debt becomes embedded in the logic of operation rather than arising solely from implementation flaws.

The next level is associated with data and models. As data volume increases and distributions change, a misalignment emerges between training and deployment. The system continues to function, but its decisions gradually lose stability. The error does not manifest as a failure; instead, it appears as a degradation in decision quality. This represents a fundamental distinction of AI systems: technical debt does not halt the system—it distorts its behavior.

Architecture reinforces this effect. As pipelines become more complex and dependencies grow, the system loses transparency. Changes begin to affect multiple components simultaneously. As a result, the cost of modifications increases, and changes are postponed. A cycle of debt accumulation emerges, in which each temporary compromise reinforces existing constraints. Similar processes are observed in complex AI pipelines and distributed systems, where unstructured dependencies transform architecture into a source of systemic risk.

A distinct role is played by the localization effect of technical debt. In distributed systems, temporary compromises are perceived as isolated decisions. However, during scaling, they propagate through hidden dependencies. Local debt becomes systemic. This leads to a situation where the system formally remains modular but effectively loses controllability.

The most critical level is associated with methodological errors. The use of correlational models in tasks that require causal analysis leads to distortions in decision-making logic. The system begins to reproduce statistical relationships as causal ones and reinforces them through feedback loops. Under scaling conditions, this evolves into a stable mechanism of debt accumulation that cannot be eliminated through refactoring or architectural optimization. Existing strategies for managing technical debt reflect attempts to address this complexity but remain limited. The most advanced approaches, such as MLOps, enable monitoring, automation, and lifecycle management of models [10]. They improve system stability and reduce operational risks. However, their primary focus is on operation rather than addressing the root causes of debt formation.

Automated detection methods, including the analysis of textual development artifacts, expand diagnostic capabilities. They make it possible to identify hidden forms of debt that are not captured in code. Nevertheless, such approaches operate post factum: they detect already existing problems without preventing their emergence. Metric-based and static methods allow for the localization of problematic areas at early stages [4], thereby reducing the risk of accumulating local defects. However, during scaling, these methods lose effectiveness, as they do not account for component interactions and fail to reflect the impact of technical debt on system behavior.

More advanced approaches, related to data governance and causal modeling, enable intervention at the deeper mechanisms of debt formation [5]. They are oriented toward preventing errors rather than correcting them. However, their application is constrained by implementation complexity and the lack of integration with other system layers. As a result, existing strategies do not form a unified management framework. They remain fragmented, focused on individual layers, and in most cases reactive. Figure 2 presents the author's model of technical debt management, in which it is conceptualized as a continuous process encompassing all system levels.

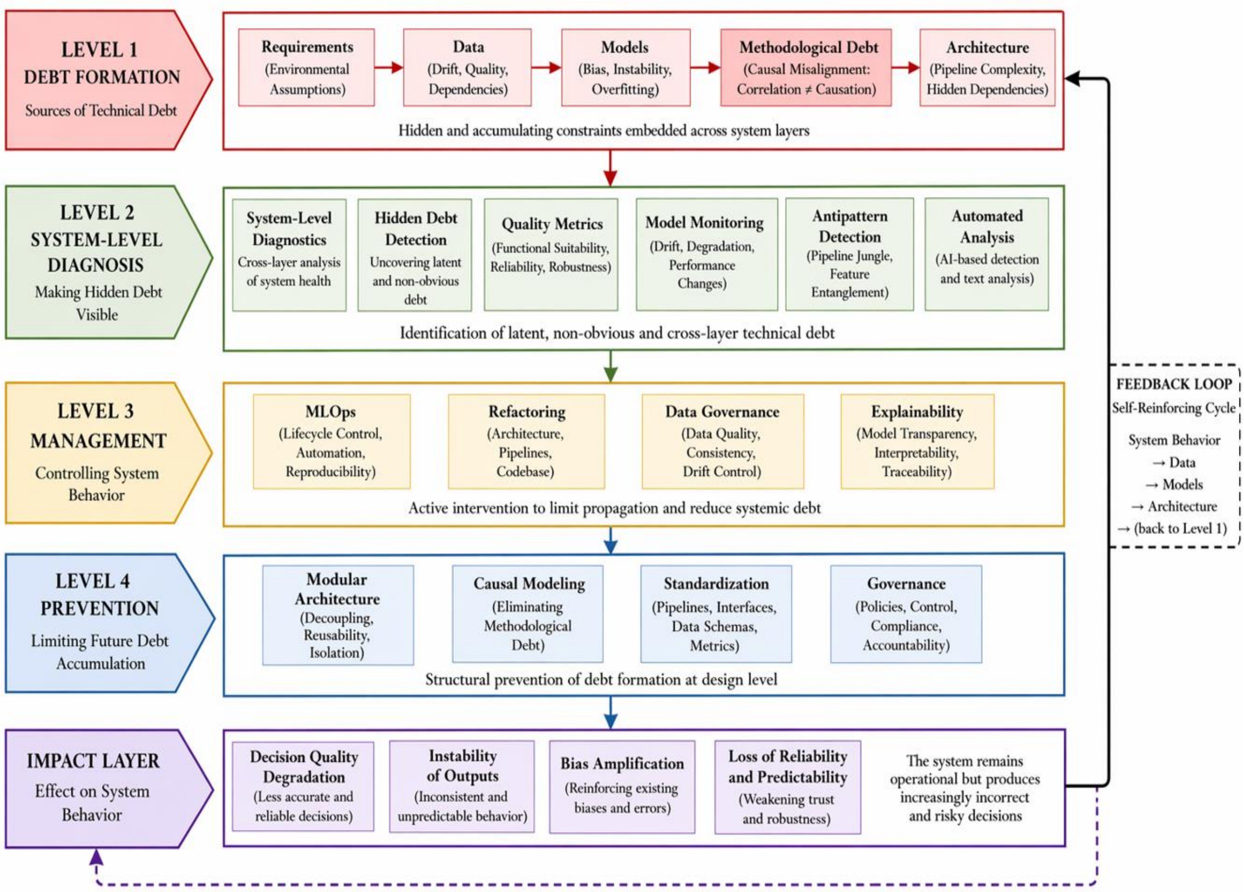

Figure 2: Technical Debt Management Model in Scalable AI Systems (Author's own development)

The proposed model is based on the premise that technical debt emerges as a result of misalignment between the system and its environment, becomes embedded in data and models, and is subsequently amplified by architecture. Within this logic, technical debt management must be integrated into the system structure rather than implemented as a separate practice. Observability is achieved through quality metrics and monitoring; however, the key shift lies in moving toward diagnostics, where technical debt is treated as a measurable and manageable phenomenon.

Management within this model is not limited to defect elimination. It is aimed at restructuring the system in a way that constrains the propagation of technical debt. This is achieved through the integration of MLOps, data governance, increased model transparency, and architectural adjustments. However, the central element remains prevention. Without it, the system inevitably returns to continuous debt accumulation.

In the proposed model, prevention is not an additional measure but a fundamental design principle. The architecture is constructed to decouple the layers of debt formation and limit its propagation. The use of modular solutions, causal modeling, and governance mechanisms reduces the system's dependence on initial assumptions and enhances its resilience under scaling.

This approach transforms the very logic of management. Technical debt is no longer viewed as a byproduct of development but becomes a factor that defines the limits of system scalability. It establishes the threshold at which the system maintains correctness, stability, and controllability.

5. Conclusion

The analysis demonstrates that technical debt in scalable AI systems can no longer be viewed as a set of local engineering trade-offs. Its formation is determined not by isolated defects but by misalignment between the system and its environment, which becomes embedded at the levels of requirements, data, models, and architecture. As a result, the core issue lies not in the existence of technical debt itself, but in how it is integrated into the system's operational logic and how it influences decision-making processes.

System stability is determined not by the volume of computational resources or the quality of individual components, but by the coherence of their interaction. Even with accurate models and sufficient data, a system may exhibit unstable behavior if decisions are made without accounting for interdependencies across system layers. When requirements, data, models, and architecture function as an integrated loop, the system maintains controllability and is able to adapt to changes without degrading decision quality.

From a practical perspective, this implies the need to shift from localized defect resolution to holistic system management. Technical debt should be treated as a continuous process requiring coordination across all levels—from design to operation. Under these conditions, the key role is played by architectural decisions, data governance mechanisms, and models, as well as tools that ensure transparency and reproducibility. The results of the study confirm the initial hypothesis that technical debt in AI systems is a multi-layered phenomenon requiring system-level management. The proposed model provides a conceptual foundation for transitioning from reactive to preventive technical debt management in scalable AI systems.

Future research directions are associated with the quantitative assessment of the impact of technical debt on AI system behavior and with validating the robustness of the proposed model under various scaling conditions. Overall, technical debt in intelligent systems should be considered not as a byproduct of development, but as a mechanism that directly determines the boundaries of their reliability, controllability, and effectiveness.

References

[1]. Alassuli, A., Eltweri, A., Thuneibat, N. S., Al-Hajaya, K., & Ismail, S. M. (2026). Artificial intelligence applications and financial forecasting accuracy in banking platforms: Evidence from Jordan. Administrative Sciences, 16(3), 122. https://doi.org/10.3390/admsci16030122

[2]. Alenazi, M. (2025). The role of environmental assumptions in shaping requirements technical debt. Applied Sciences, 15(14), 8028. https://doi.org/10.3390/app15148028

[3]. Bogner, J., Verdecchia, R., & Gerostathopoulos, I. (2021). Characterizing technical debt and antipatterns in AI-based systems: A systematic mapping study. arXiv. https://doi.org/10.48550/arXiv.2103.09783

[4]. Borowa, K., Ratkowski, A., & Verdecchia, R. (2025). The technical debt gamble: A case study on technical debt in a large-scale industrial microservice architecture. arXiv. https://doi.org/10.48550/arXiv.2506.16214

[5]. de Oliveira, N. A., & Basso, L. F. C. (2025). Advancing credit rating prediction: The role of machine learning in corporate credit rating assessment. Risks, 13(6), 116. https://doi.org/10.3390/risks13060116

[6]. Ernst, N. A., Awon, A. M., Hingmire, S., & Li, Z. S. (2026). The nature of technical debt in research software. arXiv. https://doi.org/10.48550/arXiv.2603.20415

[7]. Gordieiev, O., Gordieieva, D., Rainer, A., Gorbenko, A., & Tarasyuk, O. (2026). Quality assessment of artificial intelligence systems: A metric-based approach. Electronics, 15(3), 691. https://doi.org/10.3390/electronics15030691

[8]. Holzinger, A., Longo, L., Cangelosi, A., & Ser, J. D. (2026). Research frontiers in machine learning & knowledge extraction. Machine Learning and Knowledge Extraction, 8(1), 6. https://doi.org/10.3390/make8010006

[9]. Huh, J. (2026). Prescriptive analytics for sustainable financial systems: A causal–machine learning framework for credit risk management and targeted marketing. Systems, 14(1), 16. https://doi.org/10.3390/systems14010016

[10]. Moreschini, S., Arvanitou, E. M., Kanidou, E.-P., Nikolaidis, N., et al. (2025). The evolution of technical debt from DevOps to generative AI: A multivocal literature review. Journal of Systems and Software, 231, 112599. https://doi.org/10.1016/j.jss.2025.112599

[11]. Recupito, G., Pecorelli, F., Catolino, G., Lenarduzzi, V., Taibi, D., Di Nucci, D., & Palomba, F. (2024). Technical debt in AI-enabled systems: On the prevalence, severity, impact, and management strategies for code and architecture. Journal of Systems and Software, 219, 112151. https://doi.org/10.1016/j.jss.2024.112151

[12]. Shivashankar, K., & Martini, A. (2026). TD-Suite: All batteries included framework for technical debt classification. In A. Goldman et al. (Eds.), Agile processes in software engineering and extreme programming (XP 2026) (Lecture Notes in Business Information Processing, Vol. 578). Springer. https://doi.org/10.1007/978-3-032-22375-3_9

[13]. Simon, E. I. O., Hettiarachchi, C., Potanin, A., et al. (2026). Automated detection of algorithm debt in deep learning frameworks: An empirical study. Empirical Software Engineering, 31, 66. https://doi.org/10.1007/s10664-026-10807-5

[14]. Sklavenitis, D., & Kalles, D. (2024). Measuring technical debt in AI-based competition platforms. In Proceedings of the 13th Hellenic Conference on Artificial Intelligence (SETN '24) (Article 27, pp. 1–10). https://doi.org/10.1145/3688671.3688783

[15]. Sklavenitis, D., & Kalles, D. (2025). A scoping review and assessment framework for technical debt in the development and operation of AI/ML competition platforms. Applied Sciences, 15(13), 7165. https://doi.org/10.3390/app15137165