LATINOWARE 2025
22º Congresso Latino-americano de Software Livre e Tecnologias Abertas
Latin.Science 2025

Towards a New MLOps Architecture: A Methodological Approach Driven by Business and Scientific Requirements

Diego Nogare
Universidade Presbiteriana Mackenzie
PPGEEC - São Paulo, Brasil
ORCID: 0000-0003-0796-9431

Ismar Frango Silveira
Universidade Presbiteriana Mackenzie
PPGCA - São Paulo, Brasil
ORCID: 0000-0001-8029-072X

Leandro Augusto Silva
Universidade Presbiteriana Mackenzie
PPGEEC - São Paulo, Brasil
ORCID: 0000-0002-8671-3102

Abstract*—This article proposes an innovative conceptual model for Machine Learning Operations (MLOps) pipelines, aiming to overcome the current challenges concerning the entire lifecycle of machine learning models and to meet the growing demands of both Academia and Industry. Based on a hybrid research approach, combining scientific works and insights from professionals in the field, this proposed MLOps pipeline model integrates advanced automation, robust governance, intelligent data and model management, and explainable monitoring. We explore the convergence between theory and practice, identifying gaps and proposing an approach that promotes the scalability, reproducibility, and reliability of ML systems in complex and dynamic production environments. A state-of-the-art conceptual model for MLOps pipelines was proposed, based on a rigorous analysis of the literature and valuable insights from professional practice. The model addresses the critical challenges of automation, data and model management, monitoring, governance, and usability, aligning research ambitions with operational needs. The results from applying the MLOps architecture demonstrated measurable efficiency with a perceived improvement in the scalability, reproducibility, and reliability of ML systems. Positive outcomes were observed in relation to the deployment time of Machine Learning models, which was reduced from approximately 6 months to a range of 3 to 5 days, depending on the team's maturity and the application's purpose. An increase in productivity and operational standardization was also noted, accompanied by gains in scalability and efficiency, evidenced by the elimination of the model deployment queue, the migration of over 3,200 users to the new environment, and the publication of more than 100 Data Science models in the first few months of the new environment's operation. Additionally, the transition to a cloud infrastructure provided cost and financial resource optimization compared to the previous on-premises solution, and an enhancement of governance and security through the execution of standardized pipelines.

***Keywords*—MLOps; Methodological Architecture; Model Experimentation; Model Deployment; Model Monitoring.**

I. Introduction

The growing proliferation of Machine Learning (ML) applications has fundamentally transformed the industrial sector, driving innovations in areas such as predictive maintenance, zero-defect manufacturing, process and supply chain optimization, and real-time demand forecasting [1]. However, the transition of ML models from the research and development environment to production, especially in the dynamic scenario of Industry 4.0 and 5.0, presents complex and multifaceted challenges that extend beyond model creation [1]–[3]. Reports in the literature indicate that a considerable portion of ML projects fails to reach production or to deliver the expected value, highlighting a gap between research and the reality of industrial operationalization [2], [4].

To overcome such obstacles, Machine Learning Operations (MLOps) has emerged as a discipline that seeks to solve these problems. MLOps systematizes and optimizes the ML workflow, from experimentation to deployment, applying Developing Operations (DevOps) principles to ensure that models are developed, deployed, and maintained in a robust, reliable, and efficient manner [1], [5]–[7]. This discipline encompasses three interconnected pillars that make up the ML model lifecycle: Experimentation, Deployment, and Monitoring [8].

Experimentation covers essential activities such as data collection, analysis, and preparation; model building, training, and evaluation; as well as selection and packaging for future use [1]. This stage is intrinsically challenging due to the need to manage a growing diversity of artifacts, including datasets, features, models, and metadata, while ensuring the large-scale reproducibility of experiments [9]. Deployment, in its turn, focuses on publishing, serving and operationalizing models in various production environments, such as cloud, edge computing, and distributed ecosystems, requiring high

scalability and low latency to deal with real-time demands [10]. Finally, Monitoring is necessary for maintaining the quality and relevance of models in real-world scenarios. It covers the continuous detection of data drift and concept drift, as well as biases and feature attributions, triggering alerts that indicate the need for periodic re-evaluation or retraining of the models [1], [5], [10], [11].

Despite the recognition of its importance and the growing availability of tools, many of them open source, the full adoption of MLOps in industrial environments still faces substantial challenges [3]. A recent qualitative survey conducted with industry professionals revealed significant practical bottlenecks. Among them, the intrinsic technical complexity of data collection and preparation stands out, which complicates the automation of ML pipelines, the lack of standardization in data science development methodologies, and the difficulties in integrating with existing legacy systems in organizations, as highlighted in details at subsection III-B. Furthermore, the scarcity of professionals specialized in MLOps is a recurring concern, as are the challenges related to managing computational costs and ensuring the traceability and explainability of models [1]. The survey also pointed to gaps in available MLOps solutions, such as the need for greater ease of use, improved scalability, smoother integration with other tools, and more transparent cost management. These challenges add to the complexity of dealing with the ”hidden technical debt” that ML systems can accumulate over time [2], [12], [13].

In the face of the imperative need for synergy between scientific rigor and the practical demands of the industry, this article proposes a new conceptual model for an MLOps pipeline. Our model address the gaps and challenges identified both in the academic literature and in empirical research with professionals, offering a new methodological approach that aligns with market expectations and scientific rigor. We emphasize flexibility and the ability to integrate with the vast and growing ecosystem of open-source tools, such as MLFlow, Airflow, Kubeflow, and H2O [14], which are widely used in the industry and fundamental for continuous innovation [1], [7], [15]. By providing a methodological architecture that optimizes the ML lifecycle, promotes multidisciplinary collaboration, and ensures transparency, we aim to facilitate the operationalization of high-impact ML applications, bringing research closer to the production environment.

II. Related Works

The literature on MLOps is multifaceted, covering everything from conceptual definitions to practical approaches and implementation challenges. Following a direct approach on the main contributions of related articles, focusing on their contributions to MLOps pipelines and lifecycle automation, it was possible to identify groups of actions found in these works, namely:

A centralized ML asset platform to be a single source of truth for all ML artifacts, promoting traceability and reproducibility [9], [16]. Offering a centralized and versioned repository for features, ensuring consistency and data reuse, which solves the challenge of large tables and lack of standardization in the industry [6], [16], [17]. For challenges in storage, versioning, and management of models, including metadata and evaluation results, facilitating the management of multiple models [9]. Native integration with version control systems, for training scripts, environment configurations, as well application code [6], [18].

For intelligent pipeline orchestration, the dynamic automation and adaptation of all phases of the ML lifecycle [6]. Enabling a complete implementation of Continuous Integration (CI), Continuous Delivery/Deployment (CD), Continuous Training (CT), and Continuous Monitoring (CM), to optimize the workflow and reduce the time between stages [6], [11], [19]. Maintaining automatic mechanisms to detect data drift, concept drift, and model drift, triggering retraining routines or alerts for human intervention [10], [20], [21]. In addition to tools to ensure the quality of input data, minimizing problems at the source [10].

Comprehensive and explainable monitoring provides real-time visibility and actionable insights into the model's performance in production [1], [3], [10], [11], [13], [20]. In addition to traditional metrics, it includes Key Performance Indicators (KPIs) relevant to the business, aligning technical performance with commercial impact [22]. Also with Explainable AI (XAI) techniques such as AI Error Diagnosis Flowchart (AIEDF) [23] or Shapley Additive Explanations (SHAP) [19] to understand the complex behavior of models and identify the causes of prediction errors [23]. It also allows for the flexible creation of alerts and visualizations for the model's health and performance status [1], [3], [23].

At the governance and compliance layer, it ensures the application of responsible Artificial Intelligence (AI) principles and adherence to regulations [24]. Automatic documentation of each step and artifact, ensuring regulatory compliance and periodic audits [9], [18], [21]. In addition to the integration of human review in critical stages, such as data annotation and model validation [4], [25].

For greater adaptation and infrastructure flexibility, it offers agnostic deployment capability across cloud providers such as Amazon Web Services (AWS) through SageMaker, Microsoft with Azure ML, and Google Cloud Platform (GCP) with VertexAI, as well as on-premises infrastructures, offering flexibility and cost optimization [1], [6], [26], support for deploying and managing models on edge devices with limited resources

[1], [27], [28], as well to the use of containers (Docker) and orchestrators (Kubernetes) to standardize deployment units and manage performance [1], [21], [29], [30].

Looking for an optimized user experience, the simplification of common tasks, such as data annotation [25], model deployment, and monitoring configuration, making MLOps accessible to non-developers [25]. Tools to improve the readability of notebooks [31] and the use of semi-formal diagrams [8] for system architectures, addressing the lack of documentation and concentrated knowledge, as well as facilitating interaction and the sharing of insights among data scientists, ML engineers, and business stakeholders [24], [32].

III. Methodology

To develop the proposed model, a mixed-methods research approach was employed. First, a literature review of related works on the main topics of this research, cited in section II, was conducted to identify the state of the art of MLOps pipelines, the details of which can be followed in subsection III-A. The analysis of these articles sought to extract their main contributions, methodologies, challenges, and tools related to the automation of the ML model lifecycle. In parallel, we conducted an anonymous qualitative survey with the participation of 25 professionals from the industry, whose responses were summarized in section III-B. The objective was to gather insights on the challenges, needs, bottlenecks, tools used, governance practices, and expectations regarding MLOps in a corporate context. Finally, a synthesis and triangulation of the findings from the literature and the qualitative survey were performed. This stage allowed for the identification of gaps, points of convergence, and divergence between theory and practice, providing the basis for the development of the new conceptual model for an MLOps pipeline presented in section III-C.

A. Research methodologies from related works

The related articles were analyzed and grouped according to their research methodologies, allowing for a deeper understanding of how they operate and offer solutions to MLOps challenges.

Qualitative: Case studies [3], [6], [28], [33], [34], survey [18], thematic analysis [35], and content analysis [4];

Reviews: Individual articles apply reviews in their specific areas [6], [18], tool surveys [7], [36], and meta-analysis [7];

Empirical and experimental: Controlled studies (usability/comprehension) [8], [31], experiments with synthetic and real data [5], [23] , and experimental validation of architectures/platforms [28], [37];

Design and proposition: Proposition of methods like AIEDF [23], Before and After Correction Parameter Comparison (BAPC) [27] and SHAP [19], frameworks [5], [6], [19], [26], architectures [26], [28], model lifecycles [2], [35], platforms [17], [28], and tools [31], [37];

Technical analysis: Static analysis of code/scripts [31], [38], dependency analysis [38], profiling [37], [39], data analysis such as Principal Component Analysis (PCA) [33], and feature importance [33], multi-criteria methods [38], and A/B testing and canary deployments [17].

These articles cite or use various tools and technologies in the context of MLOps and automation, as follows:

MLOps Platforms and frameworks: Amazon SageMaker Model Monitor [10] and SageMaker Debugger [39], Valohai [7], [24], Domino Data Lab [11], OSSARA model/case [34], Edge Platform [28] e SliceOps framework [11];

Orchestration and Pipelines: Data Version Control (DVC) [9];

Version Control: Git [6], versioning tools for data, models, and scripts [18];

Monitoring: Amazon SageMaker Model Monitor [10], tools/systems for real-time monitoring [11], [17], [19], [28], [39], infrastructure monitoring for Kubernetes [29], and drift detection [5], [17];

Automation and AutoML: AutoML toolkits [17], [36], pipeline automation (CI/CD/CT/CM) [6], [34];

Development and Code: PyTorch [5], [40], TensorFlow/Keras [5], Scikit-learn [5], XGBoost [33], LightGBM [33], Pandas [38], Static Analysis tools/concepts [31], [38], HeaderGen [31], Jupyter Notebooks [31], AST parsing [31] e CFG [38];

Interpretability / Explainability (XAI): Amazon SageMaker Clarify [10] and XAI frameworks/concepts [19], [27];

Infrastructure: Kubernetes [29], AWS [10], [29], [39] and Edge Computing [28];

Experimentation: Wandb.ai [4].

And their contributions, also grouped by similarities that contemplate models, architectures, and fundamentals of MLOps, were grouped into:

Research that reviews the concept of MLOps, comparing it to DevOps, and proposes generic frameworks and pipelines or reference architectures, applied to case studies [6], [15], [34];

Research that investigates the understandability of MLOps system architectures, seeking to improve how they are documented and understood by heterogeneous teams [8], [17], [26];

Discussions that address the need to revise AI/ML lifecycle models to better suit their particularities [2], [24], [35];

Proposals for frameworks and reference architectures for Distributed Artificial Intelligence (DAI), which may involve aspects of MLOps [26], [41].

B. Qualitative survey with MLOps professionals

The qualitative survey with industry professionals revealed a detailed panorama of the real challenges and needs in the adoption and operationalization of MLOps, which were consolidated into:

Main challenges in implementing MLOps: It begins with technical and operational complexity. The management of the MLOps lifecycle is intricate, from data collection and preparation, which has high variability, to pre and post-production processes. Knowledge and people management also appear as a challenge, as MLOps knowledge tends to be concentrated in a few individuals, with other teams treating the solution as a ”black box”. There is a gap of specialized and qualified professionals to work in this discipline. The responses also present difficulties to manage large volumes of data, combined with a lack of standardization and data quality, as well as access to multiple sources and a lack of standardization in data science pipelines.

Costs: The high computational cost and the overall inherent cost of adopting MLOps are significant barriers. There is an eagerness from business leaders to have quick results in production, and the time between stages is presented as a problem. To accelerate the process, they skip steps of the MLOps pipeline and monitoring, which are often postponed to a second phase.

Main bottlenecks in MLOps adoption: This was the most consistently pointed out bottleneck; as stated in the first item, there is great difficulty in finding qualified professionals who pass rigorous selection processes. A recurring barrier, hindering automation and governance, is the integration with legacy systems, mainly because there is a lack of corporate maturity to understand the data and AI culture, thus becoming an obstacle. Complex deployment processes were also reported as a bottleneck for adoption, as many professionals who develop models do not have the specific data and ML engineering knowledge required for deployment.

Main gaps in currently available MLOps solutions: Existing tools are considered too complex, hindering widespread adoption. There are gaps in integration with other tools and applications, and respondents desire greater ease of use and integration. There is a lack of a medium/long-term and systemic vision for the models support, as well as an absence of an effective structure to obtain the ground truth and evaluate model degradation in production. Specific methods for governance and versioning of pipelines are also requested. Another point of emphasis is related to cost and Return on Investment (ROI), where there is difficulty in aligning technical feasibility with ROI and controlling costs.

Needs and expectations regarding a full MLOps methodology: There is a desire for automation of CI/CD pipelines for efficiency, reproducibility, and reliability, in order to ensure comprehensive automation throughout the cycle. There is also a need for continuous monitoring with detection of data and concept drift, and performance degradation, including business impact metrics. A lack of version control for data and models, as well as traceability for auditing, was reported as a challenge. They desire a Feature Store that is simple to use and integrated into the modeling workflow. Solutions that provide explainability and are easy to use, presenting greater integration and complementarity between MLOps, software development, and DevOps, is emphasized.

Current governance and version control practices: Most practices are still manual or heterogeneous among teams, with models often losing attention after deployment. There is use of proprietary and internal tools to catalog models and validate production authorizations. Platforms like Dataiku and MLFlow are mentioned for providing logs and explainability, facilitating governance.

C. New methodology proposed

Based on the synthesis of best practices from the literature review and the latent needs of the industry, this model was proposed to operationalize the ML life cycle in a continuous, intelligent, and data-centric manner, addressing the challenges of complexity, governance, and scalability, which can be seen in Figure 1. The model is articulated in a continuous and iterative life cycle, with the following interconnected phases:

Definition of clear business objectives [3], requirements, and ethical and governance considerations from the outset [24]

Data collection, preprocessing, data validation with validation automation [6], [10], and data ingestion into the Feature Store [16]. It includes strategies for dealing with heterogeneous and massive volumes of data.

Training, hyperparameter tuning with AutoML support [36], evaluation, and iterative versioning of models and code [9], [18]. The use of notebooks is enhanced by automatic documentation tools [31].

Automated process with CI/CD [6], [11], with support for canary deployment strategies [17], multi-environment deployment with cloud or edge [26], [28], and optimized inference management [1], [16].

Continuous detection of data/concept drift, performance degradation, bias, and anomalies [10], [20], [21].

Explainability (XAI) is fundamental to understanding deviations, and automated triggers initiate retraining or alert for human intervention [19], [23].

A transversal layer that ensures complete traceability of all artifacts and decisions in each phase [9], regulatory compliance, and information security policies.

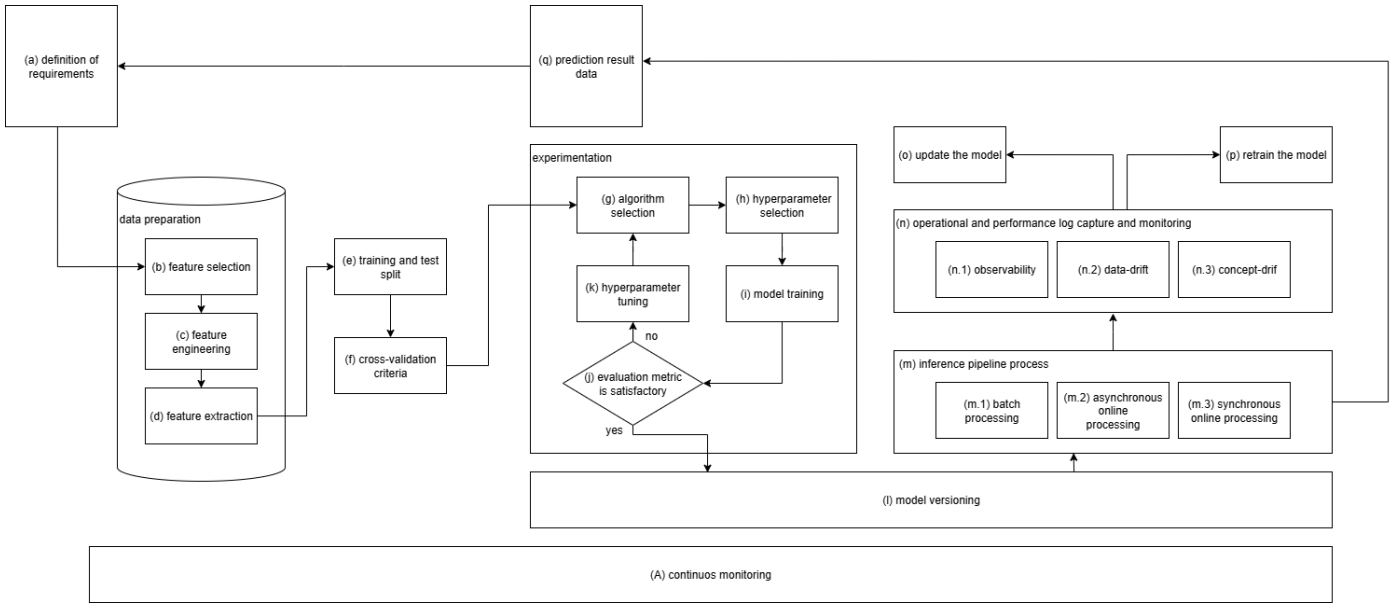

Figure 1. Journey of Machine Learning model development life cycle. Source: The Authors, 2024

Note that in Figure 1 there is the block (a) where the proposed process begins with the definition of requirements. From these requirements, data preparation is initiated, which consists of feature selection (b), feature engineering (c), and feature extraction (d). After data preparation, this dataset is spitted into a training and test subset (e), and the criterion for using cross-validation is also defined (f). After this process, the model training activity begins, which contains a cyclic process including algorithm selection (g), hyperparameter selection (h), model training (i), and checking if the model's performance evaluation metric is satisfactory (j). If it does not meet the business area's specifications, the hyperparameter tuning process is invoked (k), and the cyclic process has another iteration. However, if in activity (j) it is found that the model's performance is satisfactory, the journey proceeds to version the model (l), which will allow triggering the inference pipeline process (m). This process, in turn, can invoke one of three types of processing: batch processing $(m.1)$, asynchronous online processing $(m.2)$, or synchronous online processing $(m.3)$. After the model is promoted to production in this stage, it begins to have operational and performance log capture and monitoring (n), which has three initial validators: observability $(n.1)$, data-drift $(n.2)$, and concept-drift $(n.3)$. These metrics may, at some point, indicate the need to update the model (o) or retrain the model (p). Note that after the execution of task (m) of the inference pipeline, there is a path in the flow that leads to block (q) with the model's prediction result data. These data are used by the business area to make the decision that was defined at the beginning of this journey's cycle.

IV. Evaluation

The operationalization of Machine Learning models in production environments presents a series of complex challenges that transcend the purely scientific and mathematical domain of algorithms. Modern corporations, such as Itaú Unibanco S.A., the largest financial bank in Latin America, often face significant difficulties in managing the lifecycle of ML models, from business conception to deployment and continuous observability [42].

One of the central problems lies in the existence of multiple decentralized teams that, by using diverse technologies, greatly hinder the publication and maintenance of models in the company's information systems [42]. Itaú's original infrastructure, for example, operated entirely on-premises, which not only generated high costs but also resulted in slow development times [43]. Data scientists could wait up to six months for memory and computing resources, and the organization accumulated a waiting list of more than 100 ML models for deployment [43].

Additionally, the main difficulties included integration with legacy products, the need to build new processes for delivering Data Science models in production environments, and the adaptation of teams to new work formats [42]. The dispersion of raw data sources as System of Record (SOR) and the requirement to create specialized and centralized data sources as Source of Truth (SOT) and Specialized (SPEC) also proved to be

significant obstacles for platforms for the design, development, publication, and observability of models and software artifacts [42]. To the end, other aspects that are often neglected are related to software engineering, such as quality, performance, and reliability, in implementation and deployment in production.

Given this scenario, the architecture and components presented in the proposal of subsection III-C offer a solution to the complexity and high volume of data in Machine Learning Engineering pipelines, particularly in Deep Learning problems, ensuring the quality and scalability of the computational system. The approach integrates the three macro-phases of the MLOps project life cycle: Experimentation, Deployment, and Monitoring.

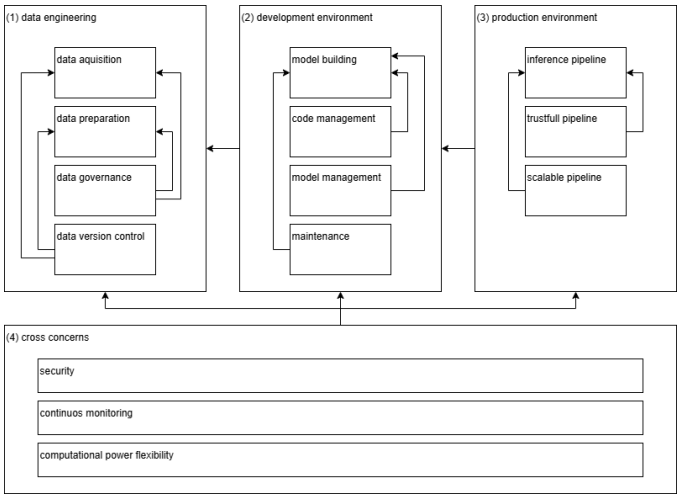

Figure 2. Solution structured with four distinct stages. Source: The Authors, 2024

The proposed architecture, illustrated in Figure 1, consists of broad platform-like designs, combining concepts of frameworks, reference architectures, and design patterns to solve the challenges of model publication. The solution is structured in four distinct stages, as presented in Figure 2, which operate in an interconnected and continuous manner, as discussed in subsection III-C

The practical application of this architecture at Itaú Unibanco, evidenced in the hybrid platform among Open-Source and Cloud Computing offerings, built in partnership with Amazon Web Services (AWS), demonstrated substantial results, validating the proposed approach [43]. The main success indicators include:

Reduction in deployment time: The time required to deploy ML models was reduced from six months to a period of three to five days. This agility in delivery boosted the bank's competitiveness and improved the customer experience [43].

Increased productivity and standardization: The standardization of the solution and better integration of ML pipelines resulted in a significant increase in the productivity of the data science team. New employees are integrated more easily, and the transfer of data scientists among departments has become more fluid [43].

Scalability and operational efficiency: The adoption of Amazon SageMaker Studio, using Jupyter Notebooks as the main development interface, and other AWS services allowed Itaú to eliminate the waiting list for model deployment. In less than a year of the new platform, more than 3.200 users were migrated to Amazon SageMaker Studio, and more than 100 Data Science models were published in the very first few months [43].

Cost savings: The migration to the cloud resulted in cost savings compared to the previous on-premises infrastructure, in addition to optimizing the use of financial resources through on-demand provisioning [42], [43].

Improved support and governance: The continuous support from the AWS team and compliance with Itaú's governance and security needs were key to the success, with governance and security issues being resolved by running pipelines to provide the platform following standardized guidelines for users [43].

It can be affirmed that the implementation of the proposed MLOps architecture, evidenced by the results at Itaú, validates its ability to transform a complex and time-consuming process into an agile, standardized, scalable, and more cost-effective operation, allowing data scientists to focus on innovation and delivering business value.

V. Discussion

The integrated and adaptive MLOps model of this study looks for to fill significant gaps among the academic state of the art and the practical needs of the industry. While the literature recognizes asset management, our model proposes to centralize all artifacts more explicitly and governed, with the Feature Store as a central piece for consistent and versioned data. This directly meets the industry's demand for better data versioning and standardization.

The emphasis on pipeline automation (CI/CD/CT/CM) with the integration of AutoML and automated drift detection differs from more segmented approaches. This addresses the industry's pursuit of efficiency and reproducibility, and the automation of complex pre and post-production processes. The inclusion of XAI and business impact metrics in monitoring is an important contribution to the industry, also including more robust monitoring for LLMs, given the absence of ground truth and for ROI visibility, which this model aims to integrate. The attention to ease of use via no-code/low-code interfaces and

the automation of notebook and architecture documentation are differentiators. The survey responses stated the need for more user-friendly tools and greater comprehensibility of MLOps knowledge, which often remains concentrated.

Although governance is addressed in the literature review of the related works, the model proposes an active governance layer that is not limited to documentary formalities, integrating automatic traceability and human-in-the-loop for greater control and compliance in real-time.

A. Comparisons with existing approaches

Existing maturity models [32] provide a path, but the proposed model offers a more detailed framework on how to advance. Commercial platforms, such as AWS SageMaker or Azure ML [6], and open-source tools like MLflow [34] and Airflow [30] are widely used, but the industry still reports gaps in their integration and ease of use. Our model suggests an architecture that maximizes the synergy between these tools, addressing the possibility of combining open-source solutions with the cloud to balance flexibility and cost. The emphasis on unifying engineering disciplines (MLOps, Software, DevOps) is a direct response to the fragmentation perceived in practice.

1) Limitations and implications: The implementation of such a comprehensive pipeline may require a significant initial investment in infrastructure and team training, especially for companies with incipient MLOps maturity. Although the model aims to simplify, the integration of multiple tools and components can still be a challenge, especially with existing legacy systems. As has been seen in recent years, the field of ML and AI is evolving rapidly, especially in matters of Generative AI, requiring the proposed model to be flexible and adaptable to new technologies and methodologies that may arise. The successful adoption of the model also depends on a cultural shift within the organization, promoting collaboration and breaking down silos between data, ML, and software development teams.

VI. Conclusion

This article has proposed and detailed a state-of-the-art conceptual model for Machine Learning Operations (MLOps) pipelines, based on a hybrid research approach that combined a rigorous analysis of scientific literature with valuable insights from professional practice. The design of this model was aimed directly at overcoming the challenges inherent in the Machine Learning model life cycle, from experimentation to deployment and continuous monitoring. The model critically addresses aspects such as advanced automation, robust governance, intelligent management of data and models, explainable monitoring, and usability, seeking to promote the scalability, reproducibility, and reliability of ML systems in complex and dynamic production environments.

The empirical validity and practical effectiveness of the proposed MLOps architecture were demonstrated through its application and validation in a large-scale scenario within Itaú Unibanco S.A., the largest financial institution in Latin America. The results of this implementation validated the capability proposed by the pipeline, showing a significant optimization that converted complex and time-consuming processes into agile, standardized, scalable, and economically efficient operations. This practical validation not only fills a critical gap among academic research and industrial needs, by mitigating bottlenecks such as the technical complexity of creating this type of solution, but also reinforces the model's contribution to facilitating the continuous delivery of strategic value through ML applications.

Despite the demonstrated robustness, it is known that the implementation of such a comprehensive MLOps pipeline may require a considerable initial investment in infrastructure and team training, especially for companies with incipient MLOps maturity. Furthermore, the continuous adaptation of the model is essential in the face of the rapid evolution of AI technologies, most recently Generative AI, and the need to promote a cultural shift towards greater multidisciplinary collaboration.

For future work, it is proposed to validate and refine the model through empirical case studies in various industrial sectors beyond finance, quantifying its impact on technical and business performance metrics. Other directions include the in-depth integration and specific monitoring for Generative AI and LLM pipelines, the creation of guidelines for the synergy between software and machine learning engineering to foster more efficient teams, and the exploration of serverless components for operational cost optimization. Such research paths aim to further consolidate the applicability and impact of the proposed model in the MLOps landscape, driving maturity and efficiency in the operationalization of artificial intelligence.

Acknowledgment

We would like to express our deep gratitude to the Instituto Presbiteriano Mackenzie for the support provided during the development of this research. The financial support, infrastructure, and resources provided were fundamental to the success of this work. We firmly believe in the importance of their contribution to the advancement of knowledge and research in our country.

Any opinions, findings, and conclusions expressed in this manuscript are those of the authors and do not necessarily reflect the views, official policies or position of Universidade Presbiteriana Mackenzie.

For the linguistic refinement of this manuscript, particularly concerning English language flow and semantic accuracy, Google Gemini was utilized as an artificial intelligence tool.

REFERENCES

[1] L. Colombi, A. Gilli, S. Dahdal, I. Boleac, M. Tortonesi, C. Stefanelli, and M. Vignoli, "A machine learning operations platform for streamlined model serving in industry 5.0," in *NOMS 2024-2024 IEEE Network Operations and Management Symposium*. IEEE, 2024, pp. 1–6.

[2] R. Ranawana and A. S. Karunananda, "An agile software development life cycle model for machine learning application development." Institute of Electrical and Electronics Engineers Inc., 2021. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9664736&isnumber=9664654

[3] L. Faubel and K. Schmid, "Mlops: A multiple case study in industry 4.0," in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2024, pp. 01–08.

[4] S. Shankar, R. Garcia, J. M. Hellerstein, and A. G. Parameswaran, "" we have no idea how models will behave in production until production": How engineers operationalize machine learning," *Proceedings of the ACM on Human-Computer Interaction*, vol. 8, no. CSCW1, pp. 1–34, 2024.

[5] J. Antony, D. Jalušić, S. Bergweiler, Á. Hajnal, V. Žlabravec, M. Emődi, D. Strbad, T. Legler, and A. C. Marosi, "Adapting to changes: A novel framework for continual machine learning in industrial applications," *Journal of Grid Computing*, vol. 22, no. 4, p. 71, 2024.

[6] R. Subramanya, S. Sierla, and V. Vyatkin, "From devops to mlops: Overview and application to electricity market forecasting," *Applied Sciences (Switzerland)*, vol. 12, 10 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/19/9851

[7] I. Zimmerman, J. Silge, P. Abedin, and R. Sanchez-Arias, "Meta-analysis of the machine learning operations open source ecosystem," in *2023 International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2023, pp. 922–925.

[8] S. J. Warnett and U. Zdun, "On the understandability of mlops system architectures," *IEEE Transactions on Software Engineering*, 2024.

[9] S. Idowu, D. Strüber, and T. Berger, "Asset management in machine learning: State-of-research and state-of-practice," *ACM Computing Surveys*, vol. 55, 12 2022. [Online]. Available: https://doi.org/10.1145/3543847

[10] D. Nigenda, Z. Karnin, M. B. Zafar, R. Ramesha, A. Tan, M. Donini, and K. Kenthapadi, "Amazon sagemaker model monitor: A system for real-time insights into deployed machine learning models." Association for Computing Machinery, 8 2022, pp. 3671–3681. [Online]. Available: https://doi.org/10.1145/3534678.3539145

[11] M. Barry, J. Montiel, A. Bifet, S. Wadkar, N. Manchev, M. Halford, R. Chiky, S. E. Jaouhari, K. B. Shakman, J. Al Fehaily *et al.*, "Streammlops: Operationalizing online learning for big data streaming & real-time applications," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 3508–3521.

[12] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, and D. Dennison, "Hidden technical debt in machine learning systems," *Advances in neural information processing systems*, vol. 28, 2015.

[13] M. Barry, A. Bifet, and J.-L. Billy, "Streamai: dealing with challenges of continual learning systems for serving ai in production," in *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. IEEE, 2023, pp. 134–137.

[14] D. Nogare, I. F. Silveira, R. Banzai, and M. C. Alexandre, "Make or buy strategy for machine learning operations–mlops," *Anais da Academia Brasileira de Ciências*, vol. 97, no. 2, p. e20240924, 2025.

[15] K. H. Chen, H. P. Su, W. C. Chuang, H. C. Hsiao, W. Tan, Z. Tang, X. Liu, Y. Liang, W. C. Lo, W. Ji, B. Hsu, K. Hu, H. Jian, Q. Zhou, and C. M. Wang, "Apache submarine: A unified machine learning platform made simple." Association for Computing Machinery, Inc, 4 2022, pp. 101–108. [Online]. Available: https://doi.org/10.1145/3517207.3526984

[16] J. De La Rúa Martínez, F. Buso, A. Kouzoupis, A. A. Ormenisan, S. Niazi, D. Bzhalava, K. Mak, V. Jouffrey, M. Ronström, R. Cunningham *et al.*, "The hopsworks feature store for machine learning," in *Companion of the 2024 International Conference on Management of Data*, 2024, pp. 135–147.

[17] I. L. Markov, H. Wang, N. S. Kasturi, S. Singh, M. R. Garrard, Y. Huang, S. W. C. Yuen, S. Tran, Z. Wang, I. Glotov, T. Gupta, P. Chen, B. Huang, X. Xie, M. Belkin, S. Uryasev, S. Howie, E. Bakshy, and N. Zhou, "Looper: An end-to-end ml platform for product decisions." Association for Computing Machinery, 8 2022, pp. 3513–3523. [Online]. Available: https://doi.org/10.1145/3534678.3539059

[18] A. Serban, K. V. D. Blom, H. Hoos, and J. Visser, "Adoption and effects of software engineering best practices in machine learning." IEEE Computer Society, 10 2020. [Online]. Available: https://doi.org/10.1145/3382494.3410681

[19] F. Rezazadeh, H. Chergui, L. Alonso, and C. Verikoukis, "Sliceops: Explainable mlops for streamlined automation-native 6g networks," *IEEE Wireless Communications*, 2024.

[20] B. Eck, D. Kabakci-Zorlu, Y. Chen, F. Savard, and X. Bao, "A monitoring framework for deployed machine learning models with supply chain examples." Institute of Electrical and Electronics Engineers Inc., 2022, pp. 2231–2238. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10020394&isnumber=10020156

[21] V. Kumar, D. Ghosh, and S. Srivastava, "Efficient mlops pipeline for transfer learning and reuse of pre-trained ml models," in *2023 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. IEEE, 2023, pp. 1–6.

[22] L. C. Silva, F. R. Zagatti, B. S. Sette, L. N. D. S. Silva, D. Lucredio, D. F. Silva, and H. D. M. Caseli, "Benchmarking machine learning solutions in production." Institute of Electrical and Electronics Engineers Inc., 12 2020, pp. 626–633. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9356298&isnumber=9356131

[23] K. Sakuma, R. Matsuno, and Y. Kameda, "A method of identifying causes of prediction errors to accelerate mlops," in *2023 IEEE/ACM International Workshop on Deep Learning for Testing and Testing for Deep Learning (DeepTest)*. IEEE, 2023, pp. 9–16.

[24] S. Laato, T. Birkstedt, M. Mäantymäki, M. Minkkinen, and T. Mikkonen, "Ai governance in the system development life cycle: insights on responsible machine learning engineering," in *Proceedings of the 1st International Conference on AI Engineering: Software Engineering for AI*, ser. CAIN '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 113–123. [Online]. Available: https://doi.org/10.1145/3522664.3528598

[25] H. Kim, B. Kim, W. Lu, and L. Li, "No-code mlops platform for data annotation," in *2023 IEEE International Conference on Memristive Computing and Applications (ICMCA)*. IEEE, 2023, pp. 1–6.

[26] N. Janbi, I. Katib, and R. Mehmood, "Distributed artificial intelligence: Taxonomy, review, framework, and reference architecture," *Intelligent Systems with Applications*, vol. 18, p. 200231, 2023.

[27] L. Fischer, L. Ehrlinger, V. Geist, R. Ramler, F. Sobiezky, W. Zellinger, D. Brunner, M. Kumar, and B. Moser, "Ai system engineering - key challenges and lessons learned," 2020. [Online]. Available: https://www.mdpi.com/2504-4990/3/1/4

[28] N. Psaromanolakis, V. Theodorou, D. Laskaratos, I. Kalogeropoulos, M.-E. Vlontzou, E. Zarogianni, and G. Samaras, "Mlops meets edge computing: an edge platform with embedded intelligence towards 6g systems," in *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2023, pp. 496–501.

[29] J. G. Almaraz-Rivera, "An anomaly-based detection system for monitoring kubernetes infrastructures," *IEEE Latin America Transactions*, vol. 21, no. 3, pp. 457–465, 2023.

[30] H. S. Kabbay, "Streamlining ai application: Mlops best practices and platform automation illustrated through an advanced rag based chatbot,"

in *2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS)*. IEEE, 2024, pp. 1304–1313.

[31] A. P. S. Venkatesh, S. Sabu, M. Chekkapalli, J. Wang, L. Li, and E. Bodden, "Static analysis driven enhancements for comprehension in machine learning notebooks," *Empirical Software Engineering*, vol. 29, no. 5, p. 136, 2024.

[32] M. M. John, D. Gillblad, H. H. Olsson, and J. Bosch, "Advancing mlops from ad hoc to kaizen," in *2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2023, pp. 94–101.

[33] E. Kannout, M. Grodzki, and M. Grzegorowski, "Considering various aspects of models' quality in the ml pipeline - application in the logistics sector." Institute of Electrical and Electronics Engineers Inc., 2022, pp. 403–412. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9908747&isnumber=9908601

[34] S. Moreschini, D. Hästbacka, and D. Taibi, "Mlops pipeline development: The ossara use case," in *Proceedings of the 2023 International Conference on Research in Adaptive and Convergent Systems*, 2023, pp. 1–8.

[35] M. Haakman, L. Cruz, H. Huijgens, and A. van Deursen, "Ai lifecycle models need to be revised: An exploratory study in fintech," *Empirical Software Engineering*, vol. 26, 9 2021. [Online]. Available: http://link.springer.com/article/10.1007/s10664-021-09993-1

[36] M. A. Al Alamin and G. Uddin, "How far are we with automated machine learning? characterization and challenges of automl toolkits," *Empirical Software Engineering*, vol. 29, no. 4, p. 91, 2024.

[37] A. Isenko, R. Mayer, J. Jedele, and H. A. Jacobsen, "Where is my training bottleneck? hidden trade-offs in deep learning preprocessing pipelines." Association for Computing Machinery, 6 2022, pp. 1825–1839. [Online]. Available: https://doi.org/10.1145/3514221.3517848

[38] L. Boué, P. Kunireddy, and P. Subotić, "Automatically resolving data source dependency hell in large scale data science projects," in *2023 IEEE/ACM 2nd International Conference on AI Engineering–Software Engineering for AI (CAIN)*. IEEE, 2023, pp. 1–6.

[39] N. Rauschmayr, S. Kama, M. Kim, M. Choi, and K. Kenthapadi, "Profiling deep learning workloads at scale using amazon sagemaker." Association for Computing Machinery, 8 2022, pp. 3801–3809. [Online]. Available: https://doi.org/10.1145/3534678.3539036

[40] H. Zhang, L. Cruz, and A. V. Deursen, "Code smells for machine learning applications." Institute of Electrical and Electronics Engineers Inc., 2022, pp. 217–228. [Online]. Available: https://doi.org/10.1145/3522664.3528620

[41] P. Ruf, C. Reich, and D. Ould-Abdeslam, "Aspects of module placement in machine learning operations for cyber physical systems." Institute of Electrical and Electronics Engineers Inc., 2022. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9797080&isnumber=9797069

[42] D. Nogare, R. F. Mello, and M. A. Lopes, "Automação no processo de publicação de modelos de ciência de dados," in *Congresso Brasileiro de Software: Teoria e Prática (CBSoft)*. SBC, 2022, pp. 40–43.

[43] D. Nogare, R. F. Mello, and V. Azeka. (2024) Itaú melhora a velocidade de lançamento no mercado e a produtividade de soluções de ml usando a amazon web services. [Online]. Available: https://aws.amazon.com/pt/solutions/case-studies/itau-ml-case-study/