УДК 004.85:004.65 DOI: 10.31673/2412-4338.2024.026874

Onyshchenko Viktoriia
University of Warmia and Mazury in Olsztyn, Olsztyn, Poland
ORCID 0000-0002-3126-2260

Vlasiuk Yevhenii
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv
ORCID 0009-0007-9036-1510

ANALYSIS OF ML MODELS LIFECYCLE MANAGEMENT EVOLUTION IN ANALYTICAL DATA PLATFORMS

Abstract: *The analysis of how principles and processes for operationalization of ML models lifecycle were changing along with evolution of data analytics platform up to Data Mesh was done in this article. Firstly, overview of ML operations in OLTP, OLAP and data lake type of platform is performed. While describing ML operations in Data Mesh, analysis of how MLOps recommendations are aligned with core Data Mesh principles is performed with identification of challenges and pitfalls. The challenge of data accessibility for ML models in cross business domains environment with respect to domain-oriented decentralized data ownership and architecture principle is highlighted as one of the most challenging and critical. In a pursuit of resolving identified challenge, article provides analysis of federated learning and feature mesh approaches comparing perspectives of each approach. Finally, conclusions and analysis of remaining and consequential challenges are represented along with topics for further research.*

Keywords: *data mesh, MLOps, ML product, OLTP, OLAP, federated learning.*

Онищенко Вікторія Валеріївна
Вірмінсько-Мазурський університет в Ольштині, Ольштин, Польща
ORCID 0000-0002-3126-2260

Власюк Євгеній Романович
Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Київ
ORCID 0009-0007-9036-1510

АНАЛІЗ ЗМІНИ ПРОЦЕСУ ОПЕРАЦІОНАЛІЗАЦІЇ ЖИТТЄВОГО ЦИКЛУ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ В РОЗПОДІЛЕНИХ СИСТЕМАХ СІТОК ДАНИХ

Анотація: *У цій статті було проведено аналіз того, як змінювалися принципи та процеси операціоналізації життєвого циклу моделей машинного навчання разом із еволюцією платформ аналізу даних аж до архітектури розподілених сіток даних (Data Mesh). Спершу виконується огляд операціоналізації життєвого циклу моделей машинного навчання в OLTP, OLAP і платформах типу озера даних. Під час опису операцій машинного навчання в Data Mesh платформах виконується аналіз того, як рекомендації операціоналізації життєвого циклу моделей машинного навчання узгоджені з основними принципами Data Mesh, з визначенням відповідних проблем. Доступність даних для моделей машинного навчання в середовищі між бізнес-доменами щодо доменно-орієнтованого децентралізованого володіння даними та принципу архітектури виділяється як одна з найбільш складних і критичних. У пошуках вирішення виявленої проблеми в статті представлено аналіз підходів федеративного навчання та продуктів даних для процесу навчання моделей, порівнюючи перспективи кожного підходу. В якості висновків, представлений аналіз проблем, що залишилися не охопленими зазначеними підходами, а також розглянуті теми для подальших досліджень.*

Ключові слова: *розподілена система сіток даних, федеративне машинне навчання, аналітична платформа.*

1. Introduction

The machine learning process was always one of the most active drivers of building and enhancing new types and architecture approaches of data platform. The quality of final ML product is directly and heavily dependent on variety, volume, completeness, accuracy of input datasets. This is a determinative factor of the success for machine learning process.

Nowadays, machine learning processes is blooming as data platforms achieved exceptionally high level of maturity and allow to provide versatile datasets, support various forms and formats of data, process huge amounts of data points within reasonable timeframe. However, it was not always the case. Data platforms were enhancing, maturing and improving for over decades to support current level of data management processes.

The main purpose of this article is to analyze how machine learning components were changing in along with analytical data platform and identify how ML processes can be aligned with Data Mesh principles. Section 2 gives overview of ML process in monolithic data platforms. Section 3 covers analysis of ML process alignment with Data Mesh principles. Section 4 describes existing solutions for data availability and access for ML in Data Mesh. Finally, section 5 discusses unresolved challenges of integration ML model lifecycle with Data Mesh principles.

2. Evolution of ML process in monolithic data platforms

2.1. ML process in OLTP platforms

Back in 1960s, first time term “database” was used. First commercial databases back in 1970s-1980s were built as supporting relational database management systems (RDBMS) and were belonging to Online Transaction Processing (OLTP) platforms [1]. Cost of storing a megabyte of data in those years was high so each decision about storing a particular data point was very well-balanced. Data-hungry machine learning algorithms didn’t have an ability to work with large datasets due to this reason as well. Relational model of data persistence was also not perfect match for machine learning algorithms which by nature expect intensive read operations over the big dataset. RDBMS structure with data normalization and deduplication, intensive usage of transactions, isolation levels and records lock weren’t primarily designed and optimized for intensive read operations, while main focus was on writing and storing data supporting consistency and concurrent environment with parallel execution.

These factors were not contributing to ML models of those time. Limitations on data-read operations as well as thoughtful storage were forcing ML models to limit data consumptions and look for permanent balance between cost and performance of the platform. To keep the last on a proper level, ML models training process was offloaded to separate schema or database instance via data-replication process but cost efficiency was still a concern for such approach.

2.2. ML process in OLAP platforms

Late 1980s, beginning of 1990s brought a shift of paradigm: term “business data warehouse” (DWH) was introduced along with building first database which belongs to Online Analytical Processing (OLAP) type of platforms. OLAP platforms brought new multidimensional data model and prioritized data read operations over data write operations. They started to store pre-aggregated data in a form suitable for data consumers: business intelligence reports, analytics and for sure - machine learning. The discipline of data analysis got special attention as analysts occurred in a situation when they needed to work with wide variety of data in various models, forms and formats [2].

ML algorithms got a source of data which they were be able to consume to feed high volume of data to run intensive training process of the ML model without high latency, delay and impact to the operational transactional application. For sure such benefits don’t come without additional cost and overhead. OLAP platforms were not a substitution for existing OLTP relational databases. They became additional citizens in the overall data platform of the organization. OLTP databases as earlier were handling data persistence from the transactional applications. As a separate process detached from the main transactional workflow, data preparation and population to OLAP platform was establishing. It allowed to pre-aggregate and prepare data model in a format which will be the most sufficient for consumers, including machine learning algorithms and product. Once process of transforming data and loading it to OLAP platform was done, machine learning applications were

able to read necessary amounts of data for training, validation or inference. Such approach significantly increased data availability to various consumers, however, brough data duplication between OLAP and OLTP platforms and latency for the data points in OLAP platforms.

OLAP platforms in 1990s brought paradigm shift from prioritizing write operations to prioritizing read operations. However, they still were operating within relational model representing data as tables, rows, columns. Decade from 2000 was full of new databases grounded on non-relational model. First XML-based database was released, timeseries, graph-oriented databases became popular as well during same years. Increased amount of data formats supported by data platforms were beneficial for machine learning algorithms. Now those were able to consume data not only in tabular formats from OLAP platforms but also from XML or JSON files, graphs, etc. It became obvious that it's not always necessary to represent data in relational form so more and more data points were kept in raw form and stored in text-based files. Velocity of collecting and persisting data increased a lot. While it was convenient for some of machine learning algorithms, the general data availability for end users decreased and abilities for vertical scaling ran out. It produced a demand for platforms which would allow to process massive number of data files stored on multiple machines within reasonable time. That was the time when first version of Hadoop platform was released which changed again the landscape of the data platforms in the organizations.

2.3. ML process in data lake platforms

Hadoop gave an instrument to operate with raw data stored in files format on large number of computational machines. It opened an ability for horizontal scaling of data platforms storing more data without limit of computational or storage resources of individual machine.

ML algorithms got an ability to consume huge variety of data formats, from texts and timeseries, ending up with pictures and video files. As OLAP platforms earlier complemented OLPT platforms and enabled intensive data-read applications, Hadoop extended architecture of data platform with new components. Now data from OLTP platforms and other data producers were collected in Hadoop and stored in raw format. Horizontal scaling of Hadoop and effective data processing with MapReduce and Spark allowed to collect in one platform data from various departments of the organization and build cross-divisional data products. It started a concept of "data lake" which became popular and well-adopted in many organizations. OLAP platforms became a target persistent platform for such pre-aggregated data products. Machine learning algorithms got one more source of data for its operations.

During last decade, architecture of modern data platform was mainly combining OLTP platforms, data lake, data warehouse, streaming platforms, operational layer with significant focus on data governance, security and adoption of cloud computing. Major cloud computing providers developed their cloud-native platforms for data lakes and data warehouses. A couple of variations of data platform architecture were developed and widely adopted: data factory, data fabric, data Lakehouse [3].

Machine learning was always a first-class citizen of such platforms and architectures.

Operations of machine learning algorithms includes feature engineering from data persisted in data lake or data warehouse, model training, versioning and management, ending up with model inference and continuous monitoring with data and model drift detection [4]. Model and data lifecycle management is ruled by MLOps principles applied in the organization.

Modern data architectures like data fabric or lakehouse solved long-lasting problem of collocating data and gave an ability to build cross-organization analytics. Finally, finance data and marketing data are living in one data lake and organization can analyze them together with no silos in data as earlier. However, rapid speed of collecting new data quickly brought a situation when single data lake for the organization became huge and highly complex platform which requires expensive management and support. Single monolithic data lake complicates appliance of data quality and data governance rules which are different for different business domains of the organization as nature of the data is different [5]. That's how in 2020, concept of Data Mesh started which is an example of distributed data platform architecture. Data Mesh dictates own rules and principles [6] among which are strict usage of data product by the consumers in different domains and avoiding of using raw data

outside of business domain. It's brining new challenges for machine learning which tend to intensively use raw data collected in one data lake from various domains.

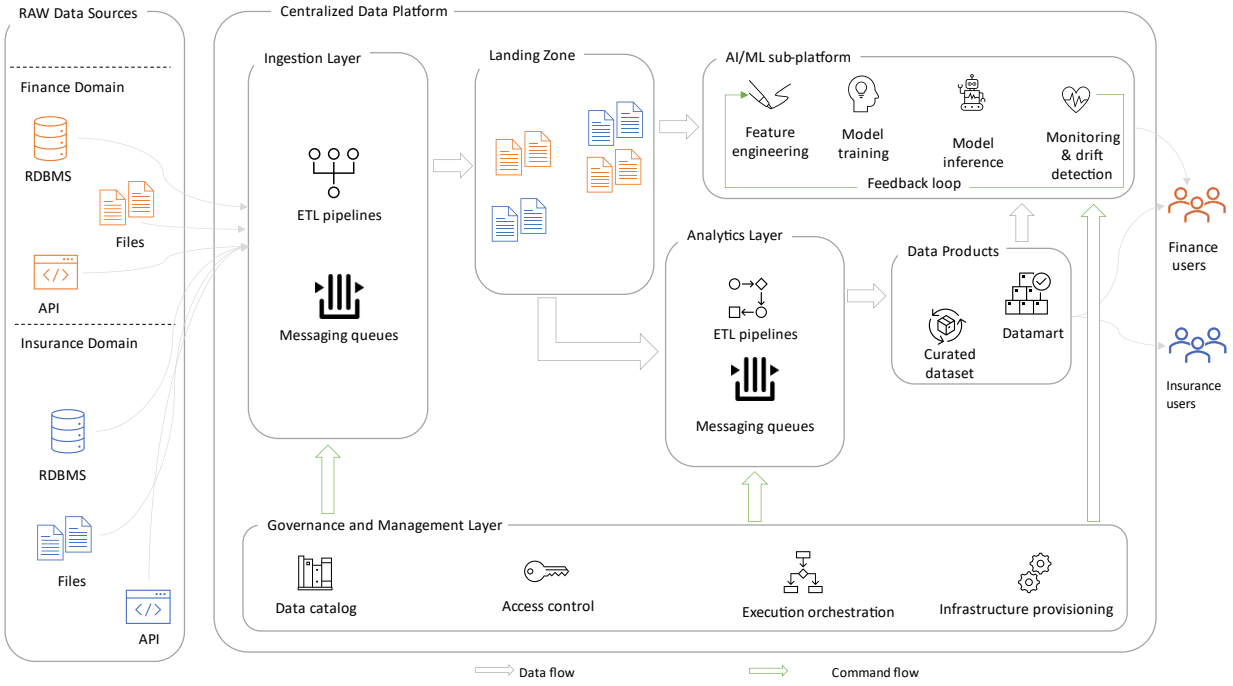

Fig. 1. Machine learning components in modern data platform

3. MLOps process alignment with Data Mesh principles

There are a lot of use cases for the MLOps process setup across multiple organizations in their ultimate goal of building multi-faceted ML products. Tuomas Granlund, Aleksi Kopponen, Vlad Stirbu, Lalli Myllyaho, Tommi Mikkonen in their research [7] analyzes challenges of scaling AI/ML operations to multi-organization landscape. Data Mesh can be represented as architecture pattern for analytic platform of the organization with multiple business domains which will in fact correlate to multi-organization setup as each domain can be easily represented as separate independent organization. Moreover, quite frequently, large enterprises are growing by active acquisitions bringing new companies with their corresponding data platforms under one organization and one umbrella of data analytics landscape. Saying that, we want to bring the idea that challenges in adopting MLOps in Data Mesh platform are highly correlating with challenges in multi-organizational setup.

Currently MLOps is focusing on describing how reliably manage lifecycle of the models up to release of ML products [8]. However, it doesn't mention a lot about scaling towards the whole organization, especially multi-domain organizations. Data Mesh in comparison vice versa focuses on data analytics landscape management on enterprise level focusing on scalability questions. Naturally MLOps principles should be aligned and accommodated and Data Mesh principles.

Data as a product principle of Data Mesh recommends to treat end-user data as software product and apply continuous monitoring, maintenance and improvement as to any software product. MLOps is aligned with this as also requires appliance of same processes towards ML models. MLOps feedback loop component allows continuous enhancement of the model in line with Data Mesh recommendations for data products.

Self-serve data infrastructure as a platform principle aims to bring automation in every place of data analytics platform to minimize operational overhead for creating and maintaining such platforms for each business domain of the organization. MLOps also heavily relies on automated pipelines for model training, deployment, and monitoring, enabling domain teams to self-serve these processes without heavy reliance on central IT or data engineering teams.

Federated computational governance principle advocates for implementation of governance policies in a decentralized manner but at the same time coordinating across multiple business domains. As we were stating earlier, MLOps is currently focused on model lifecycle management rather than scaling on organization level so majority of MLOps processes should be aligned and added to federated governance layer of data mesh platform.

Finally, domain-oriented decentralized data ownership and architecture principle from one angle is well aligned with MLOps model ownership and autonomous model development recommendations where each ML model is owned by the domain team responsible for its development and deployment, however, data isolation between domains and interactions via data products introduces challenges for MLOps and ML models which shine the best in environments with high variability of data. Access to data from various domains is a one of the key challenges of scaling MLOps to multi-domain organizational level as from one end ML models need to have access to as many data points as possible with a caveat of maintaining proper data quality and governance, from the other end this pursuit might jeopardize proper cross-domain interaction. In the next section, we will analyze two approaches of aligning these challenges together.

4. Operationalization of ML products in Data Mesh

Data Mesh is not only about the software architecture of the data platform for the organization. It's also about the governance model and structure of the organization. Without properly structured data teams for the identified business domains, it will be hard to design and develop Data Mesh for the organization. Teams which are working on ML models development, training and production also should be aligned with business domains. Alex Buck, Tobias Zimmergren, Regan Downer and Liz Casey in their article [9] mention that some organizations threat such ML teams as data consumers. However, in this case ML team will need to work with data products which belong to various business domains being detached from any of them. It will cross out the main idea of Data Mesh to keep ownership and responsibility of data in teams which are subject matter experts and will lead to varied treatment of the data. Authors propose the concept of domain ownership feature engineering by domain teams to mitigate this problem.

To keep business-domain oriented ownership over the data, ML team can be part of the domain team. It shouldn't be detached from the business domain and should contain subject-matter experts about data products produced by the domain team. This organizational structure fits well for the cases when ML products built on top of data which belong to single domain. In this case, models can be trained and tested on feature sets produced on top of the domain-specific data. However, such cases are rather rarity than a rule. More usual case when model is built using data from multiple business domains. To build demand forecasting ML product in marketing business domain, CPG company needs to use data from marketing, financial, manufacturing and other business domains. Marketing data team builds and owns this ML product however can't reference raw data from other business domains which was acceptable in centralized data platforms.

Haoyuan Li in work [10] proposes an approach of Federated learning which decentralize the process of model training. It gives an ability to split model training process between data domains in Data Mesh and avoid necessity for cross-domain data usage. Author analyzes horizontal, vertical and split federated learning, compares them to understand pros and cons of each approach and proses a framework to implement federated learning as part of Data Mesh. Following federated learning approach, each domain data team should build learning process of the model on the data which belongs to their business domain. As a result, each domain data team will have partial model trained on their domain data. All partial model afterwards are combined into one Global model which is a foundation for cross-domain ML product.

Adoption of federated learning in data mesh is perspective and allows to solve main issue of cross-domain data reference for model training, however, federated learning has number of limitations among which is inability to apply for models that need to perform complex joint operations on the entire dataset, or models that require global normalization. Complexity of building real-time ML products will be also higher as federated learning involves a process of training local models and then aggregating them, which can take additional time critical for real-time use cases.

Alex Buck, Tobias Zimmergren, Regan Downer and Liz Casey in addition to the concept of domain ownership feature engineering by domain teams also propose the idea of feature mesh which can be vital addition to federated learning. Authors propose to treat feature sets used for model training as data products in line with other data products. In this case, feature set becomes a contract of communication between domains in Data Mesh, specifically for ML product use cases. In a same manner when domain data team uses data product of another domain to build own data product, feature set can be used to train model which belongs to completely different business domain.

In above example with marking data team building ML product for demand forecasting, this team can use feature sets from financial and manufacturing domains which upfront were built, curated, verified and published by data domain teams of corresponding domains. Authors propose to publish feature sets along with other data products to data catalog.

This approach at the same time opens a lot of further challenges to be solved while aligning MLOps principles with data mesh. Introducing feature sets as data products will require their proper maintenance, in-time update, curation and governance. Handing data drift brings next level of challenges as now feature set is used by multiple consumers and for one of them current version of features might not be sufficient and it might require rebuild of feature set with up-to-date data while other consumers still can work with existing features. It pushes towards versioning and backward compatibility aspects of features. Feeback loop itself will require transformation in own implementation taking into account distributed nature of the data mesh platforms. Saying this, we want to highlight that amount of research in a field of adopting MLOps in Data Mesh platforms is decent and require further work.

5. Conclusions

Build and operations of ML models changed significantly with evolution of data platforms. Nowadays, one of the challenges of ML model lifecycle management is scaling corresponding principles on enterprise level, especially multi-domain enterprise level.

Data Mesh as architectural pattern focuses on data analytics platform organization exactly for multi-domain enterprises. Naturally that MLOps principles should become aligned and integrated with main Data Mesh principles. However, distributed nature of data mesh platforms makes this alignment complicated. In this article we've analyzed how MLOps principles should be adopted to domain-oriented decentralized data ownership. We've seen solutions to overcome main problem of cross-domain data availability for data-demanding ML models but at the same time we've discovered more challenges and blind spots in processes and architecture components which are yet to be solved.

In further research, we will work on proposing software components and solutions which will help to meet the demand of data for ML models operations while keep respecting main data mesh principles of cross-domain data management.

References

1. Stavros Harizopoulos, Daniel J. Abadi, Samuel Madden, and Michael Stonebraker. OLTP through the looking glass, and what we found there. Making Databases Work: the Pragmatic Wisdom of Michael Stonebraker. Association for Computing Machinery and Morgan & Claypool. 2018. P. 409–439.

2. Forresi, C., Gallinucci, E., Golfarelli, M. et al. A dataspace-based framework for OLAP analyses in a high-variety multistore. The VLDB Journal 30, 1017–1040 (2021).

3. Michael Armbrust, Ali Ghodsi, Reynold Xin, Matei Zaharia. Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics, 2020 Retrieved from https://www.databricks.com/wp-content/uploads/2020/12/cidr_lakehouse.pdf.

4. D. Kreuzberger, N. Kühl and S. Hirschl, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," in IEEE Access, vol. 11, pp. 31866-31879, 2023.

5. Vlasiuk, Y., Onyshchenko, V. (2023). Data Mesh as Distributed Data Platform for Large Enterprise Companies. In: Hu, Z., Dychka, I., He, M. (eds) Advances in Computer Science for Engineering and Education VI. ICCSEEA 2023. Lecture Notes on Data Engineering and Communications Technologies, vol 181. Springer, Cham.

6. Dehghani Z. How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh. 2019. / Retrieved from: https://martinfowler.com/articles/data-monolith-to-mesh.html

7. T. Granlund, A. Kopponen, V. Stirbu, L. Myllyaho and T. Mikkonen, "MLOps Challenges in Multi-Organization Setup: Experiences from Two Real-World Cases," 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN), Madrid, Spain, 2021, pp. 82-88,

8. Dominik Kreuzberger, Niklas Kühl, Sebastian Hirschl. Machine Learning Operations (MLOps): Overview, Definition, and Architecture. 2022.

9. Alex Buck, Tobias Zimmergren, Regan Downer and Liz Casey Operationalize data mesh for AI/ML domain driven feature engineering. 2023. / Retrieved from: https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/scenarios/cloud-scale-analytics/architectures/operationalize-data-mesh-for-ai-ml

10. Haoyuan Li Empowering Data Mesh with Federated Learning. 2023. / Retrieved from: https://www.diva-portal.org/smash/get/diva2:1787862/FULLTEXT01.pdf