Practices for Managing Machine Learning Products: A Multivocal Literature Review

Isaque Alves, Leonardo A. F. Leite, Paulo Meirelles, Fabio Kon, and Carla Silva Rocha Aguiar

***Abstract*—Machine learning (ML) has grown in popularity in the software industry due to its ability to solve complex problems. Developing ML systems involves more uncertainty and risk because it requires identifying a business opportunity and managing source code, data, and trained models. Our research aims to identify the existing practices used in the industry for building ML applications and comprehending the organizational complexity of adopting ML systems. We conducted a multivocal literature review and then created a taxonomy of the practices applied to the ML system life cycle discussed among practitioners and researchers. The core of the study emerged from 41 selected posts from the grey literature and 37 selected scientific papers. Applying Initial Coding and Focused Coding techniques into these data, we mapped 91 practices into six core categories related to designing, developing, testing, and deploying ML systems. The results, including a taxonomy of practices, provide organizations with valuable insights to identify gaps in their current ML processes and practices and a roadmap for improving, optimizing, and managing ML systems.**

***Index Terms*—Machine learning (ML), management of scientists and engineers, multivocal literature review (MLR), practices, product life cycle, software engineering.**

I. Introduction

MACHINE learning (ML) integrates artificial intelligence (AI) capabilities into software and services [1], solving complex real-world problems. Recent advances in ML models and frameworks have led to increased adoption in the software industry, reflecting the effectiveness and profitability of these systems for companies. However, the lack of understanding about the complexity of the product development life cycle can impact the ML system delivery [1], [2].

Manuscript received 26 January 2023; revised 7 June 2023; accepted 14 June 2023. This work was supported in part by the General Secretary Ministry of the Brazilian Presidency of the Republic through the "Free Software Ecosystem for Digital citizen participation" Project, in part by the National Science and Technology Institute of the Future Internet for Smart Cities funded by the Brazilian National Council for Scientific and Technological Development under Grant 465446/2014-0, in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior Brasil under Finance Code 001, and in part by the São Paulo Research Foundation under Grant 14/50937-1, Grant 15/24485-9, and Grant 19/12743-4. Review of this manuscript was arranged by Department Editor G. Marzi. *(Corresponding authors: Paulo Meirelles; Leonardo A. F. Leite.)*

Isaque Alves, Leonardo A. F. Leite, Paulo Meirelles, and Fabio Kon are with the Department of Computer Science, University of São Paulo, São Paulo 05508-220, Brazil (e-mail: isaque.alves@ime.usp.br; eofl@ime.usp.br; paulormm@ime.usp.br; kon@ime.usp.br).

Carla Silva Rocha Aguiar is with the Faculty of Gama, University of Brasilia, Brasilia 70910-900, Brazil (e-mail: caguiar@unb.br).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TEM.2023.3287759.

Digital Object Identifier 10.1109/TEM.2023.3287759

ML systems are data driven. While standard software applications are deterministic, ML models are probabilistic [3], [4]. In other words, developing an ML product involves more uncertainties and risks throughout the product life cycle and brings new engineering, management, and organizational goals. ML models can generate rules based on the patterns inferred from data, aiming to improve the performance of tasks, predict business value to their users, or, depending on the application, make the most appropriate decision for the business context [2], [5].

From managers' perspectives, making it ML-ready and identifying business processes that could benefit from ML should anticipate ML adoption [6], [7]. Actions before ML system adoption include building teams to attack more complex problems with ML and familiarizing managers with ML techniques to ensure that their ML system requirements create maximum value for their organization [8].

The ML model entirely depends on the quality and quantity of data [9], [10]; cleaning, labeling, and training demand time and resources [11], [12]. ML product workflows necessitate a more robust orchestration of environments once there is potentially one test environment for each model hypothesis. Independent versioning and pipeline of the source code, model, and training data guarantee a repeatable deployment process [13]. In this context, MLOps is an adaptation from DevOps engineering culture and practices to enable continuous delivery of ML products [14], [15]. Finally, model monitoring in the production environment is obligatory to enable continuous model improvement, prevent significant changes in the data distributions (concept drift), and detect model performance degradation [13], [15].

Because an ML system life cycle differs from the standard software life cycle, the techniques and practices must be adapted to suit the particularities that involve the relationship between the data, the trained model, and the source code [9], [10]. Since the publication of Google's paper on the challenges of maintaining and evolving ML models in the production environment in 2015 [16], industry efforts have concentrated on identifying and developing these practices.

Our research maps the existing practices and methods discussed in both academic and practitioner literature throughout the entire life cycle of ML systems, from business opportunities to maintaining and evolving models in production environments. We conducted an empirical study in two stages, performing a multivocal literature review (MLR) using Initial and Focused Coding to analyze the data systematically. We used Initial and Focused Coding to analyze the data, map the practices, and group them into categories.

We searched publications not controlled by academic publishers, also known as grey literature (GL), including working papers, white papers, technical reports, blogs, videos, and web pages [17]. Due to its novelty, the velocity with which organizations adopt ML products, and the longer publication process of peer-reviewed academic literature, publications not indexed by scientific repositories give a vast amount of up-to-date and emerging information regarding the theme. They share partial results, experience reports, case studies, tools, and techniques. Although GL usually has less credibility than formal academic production, it also has advantages; GL is a leading source for identifying gaps not yet covered by academic literature and investigating more up-to-date and emerging phenomena [17]. Moreover, GL is more suited to study the industry perspective since practitioners publish their ideas in blogs, reports, and informal publications. Extensive research data are available as open data,[1] so other researchers can verify and extend this work in new directions.

We aim to provide an overview of ML system practices throughout the life cycle and highlight the concepts most discussed by managers, data scientists, and engineers. We present one conceptual map for each stage of the ML product life cycle, from problem/hypothesis definition to delivery/runtime stages. Understanding the practices, the roles, and their impact on the final ML system provides managers with an overview of the complexities associated with adopting ML components and the particularities that differentiate an ML system from deterministic standard software. Furthermore, we present these practices grouped in categories related to the roles or project phases, which might guide engineers, data scientists, managers, and researchers in identifying gaps. From a theoretical point of view, this research contributes to a novel conceptual map, including technical and nontechnical practices for managing an ML product throughout its entire life cycle. The conceptual map combines practices discussed in academic and practitioner literature, usually in themed and focused communities. We also contribute by identifying gaps in the literature and topics discussed in practitioner communities, revealing research opportunities. From a practical point of view, the conceptual map can guide organizations in identifying the practices that are required to adopt AI technologies successfully. This article also presents practical implications for managers, engineers, academics, and data scientists. By exploring these aspects, we aim to advance our understanding and to provide valuable insights that can guide professionals and organizations in effectively navigating the challenges and maximizing the benefits associated with AI implementation.

Moreover, previous works are focused on practices of a particular phase of the ML system, such as business [6], project development cycle [1], [18], or a particular technical challenge [19], [20]. In our work, we mapped or discussed the entire life cycle of ML products, from business opportunity to product management to model deployment. It enables us to illustrate the complexity of the interfaces between heterogeneous skilled professionals in the process and might guide managers throughout ML system adoption.

The rest of this article is structured as follows. Section II provides an overview of ML systems, including the workflow and adoption challenges. Section III focuses on the research methodology employed in this study. The results and discussion are presented in Sections IV and V, respectively. Sections VI and VII outline the implications and research opportunities stemming from this research. Section VIII discusses the existing work on ML products. Section IX addresses this study's limitations. Finally, Section X concludes this article.

II. Background

This section presents an overview of the main concepts related to managing ML products. We present the fundamental differences between standard software development and ML system development. In addition, we delve into the multidisciplinary process of ML product development, along with the associated adoption challenges.

A. ML System

In standard software, development is a set of rules and instructions that, together with the data, is intended to produce the expected deterministic results. ML systems are predominantly developed based on existing data and, in some cases, desired results that are the input of an ML algorithm [3]. They are not exclusively based on rules developed by humans. The ML system output is probabilistic and infers patterns identified in the data, instructions, and pertinent rules.

The ML system is, therefore, more complex than standard software [21]. In addition to having source code that executes instructions necessary for the system to function, there are two additional artifacts to the ML systems: the trained model and the training data. Managing and maintaining independent delivery pipelines for source code, models, and data are essential to guarantee the accuracy and reliability of the ML system in the production environment.

ML through the lens of business capabilities supports three business needs [22]: automating business processes, gaining insight through data analysis, and engaging with customers and employees. Most ambitious ML projects encounter setbacks and failures. More than replacing human capabilities, organizations must understand which technologies perform what type of tasks, create a prioritized portfolio of projects based on business needs, and develop plans to scale up across the company. ML initiatives aim to improve existing products, pursue new markets, and make better decisions.

B. ML System Development Processes

There are many possible workflows and process models for developing an ML system. Team Data Science Process (TDSP) [23], Data-Driven Scrum (DDS) [3], and Cross Industry Standard Process for Data Mining (CRISP-DM) [14] are some examples. They are all data-centric processes with continuous feedback loops [1]. Most organizations adopt one of these

[1] https://github.com/alvesisaque/Practices-ML-Product

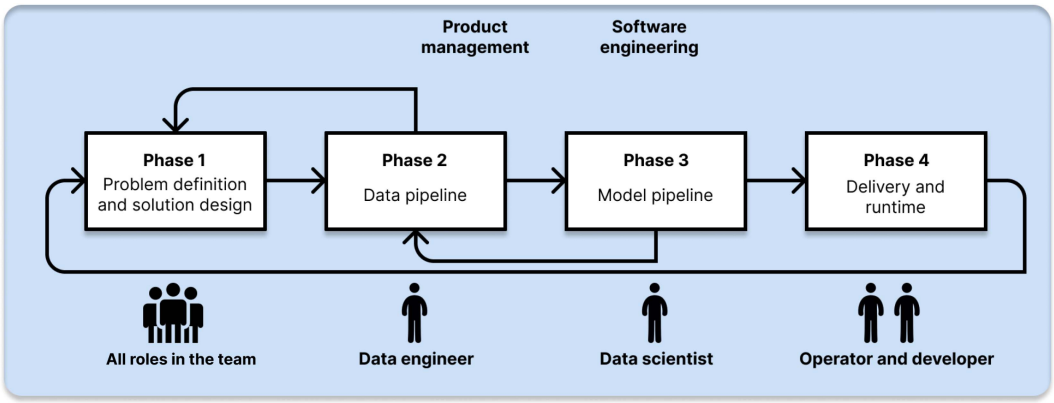

Fig. 1. Four phases of developing an ML system. The development consists of a cycle that allows feedback loops at any stage. In Phases 2 and 3, the presented feedback loop arrow aims to ensure greater alignment between model training, data selection, and solution design.

process models with agile practices. This work aims to bring the entire life cycle of an ML product more broadly, covering the roles of managers, data scientists, software engineers, and designers. Therefore, based on the already known models, we inserted processes from product engineering, such as problem understanding, solution design, and product management, besides software engineering [24], as they are fundamental in general software product development.

As presented in Fig. 1, the goal of the first phase of an ML product life cycle is crucial to deeply understand the business problems and identify which functionalities are feasible to implement with ML. The power of ML arises when the rules are not precise or are not known, and there are available datasets [1], [4]. Two questions must be answered.

1) What is the problem to be solved, and what are the expected outputs?
2) Do we have access to enough data?

In addition, other requirements addressed by ML are very complex logic, rapid scalability, the need for specialized customization, and adaptation in real time. If the product falls into any of these categories, the need to develop an ML product becomes more evident [25].

Still in the first phase, but now focused on solution design, it is essential to check the role of ML in the product, whether it will be the core of the application or support an extra feature. Preliminary software architecture design details how models will interact and if they have strict module boundaries between ML components and software engineering modules, with independent versioning and pipelines [16], [26]. These decisions will guide the team structure needed for the next phase and prescribe the models' specific intended behavior.

Phases 2 and 3 consist of developing the product specified and designed previously.

Data science and mining workflows are adapted to establish the activities necessary to develop ML functionalities, such as CRISP-DM [27]. ML products have three main pipelines: code, data, and model. Engineers collect, clean, and organize data to prevent the model from learning from data that is "dirty," biased, or could hinder learning. This phase requires care and can cost time and resources to guarantee product quality (e.g., data annotation) [6], [12]. Data scientists lead the model construction and training, using techniques such as feature engineering [1], [10] or reusing code and features [4]. As in the standard cycle of product engineering development, this phase is also closely linked to the solution design. New challenges and discoveries may arise during training, and data analysis must be considered. Release engineers configure Continuous Integration and Deploy pipelines to test and validate data, data schemas, and models [15].

The delivery and runtime monitoring of ML products (Phase 4) can also be challenging; production data must be consistent with training data, while new data should be collected to maintain the product's correct functioning [1]. Unique to ML systems, continuous training automatically retrains and serves the models, which can decay in more ways than conventional software systems [15]. Production data are mutable; therefore, continuous success measures are necessary to slightly adjust different models over time [16], [28]. The components intricately influence each other as they can have tight couplings, resulting in a complex entanglement of components (nonmonotonic error propagation), troubleshooting, and debugging [28]. Another critical point is that this product requires more robust processing to ensure agility in a prediction or continuous learning model. Tracking user feedback and continuous monitoring of the model outputs are essential to iterate the product in the subsequent cycles of development and improvement [1], [9], [29].

Fig. 1 presents product management and software engineering aspects. They are outside the represented cycle as they are present throughout the entire process: strategic, conceptual, operational, developmental, and project evolution stages [30]. The product manager is responsible for ensuring that the results obtained align with the expected results of the stakeholders. This role ensures product execution and success, taking care of technical and business aspects such as planning and defining the roadmap, strategic management, and keeping the project in line with the business objectives [31]. On the other hand, software engineering also plays an essential role in an ML system process and quality. Aspects such as verification and validation, alignment between the real world and the validation set, the architecture definition to meet the problem, and the

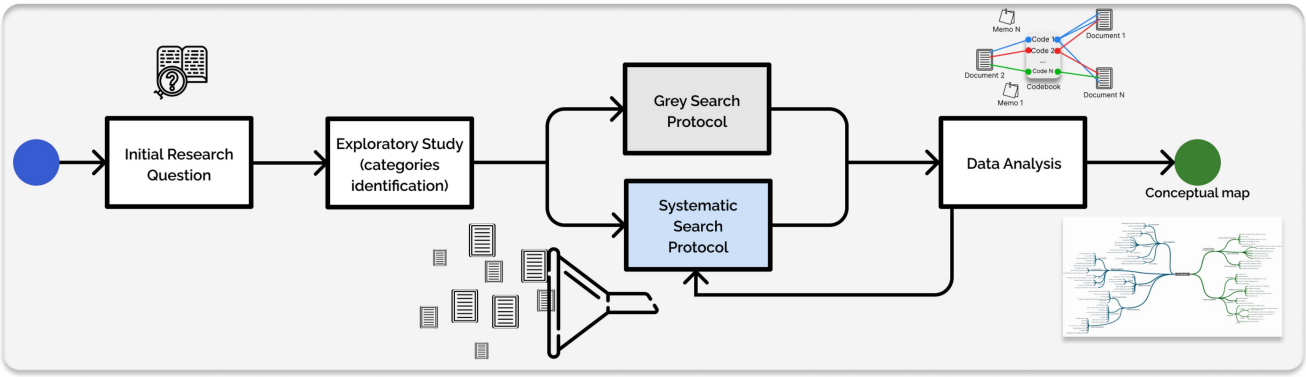

Fig. 2. Methodology key stages; the grey and systematic search protocols are described in Figs. 3 and 6, respectively. The arrows indicate that if the theoretical saturation [34] is not achieved, researchers should conduct new searches in the literature review process, primarily using reference snowballing.

proposed solution are knowledge that the software engineer shares with the team to build the environment and design the software. Automation, deployment and continuous integration, environment control, and versioning are strong points when adding software engineering to an ML project [1]. Also, in recent years, it has been possible to notice an effort to build frameworks and environments to standardize and operationalize the ML life cycle process and aspects related to ISO standards [32], [33].

C. Adoption Challenges

About adoption challenges, for instance, testing the trained models over time, with constantly evolving data, is challenging [35]. Mikkonen et al. [36] explain that the results can be stochastic, statistical, or evolutionary over time. That is, while they are correct, errors may not be identified.

Building a product that effectively meets the needs requires addressing various challenges, including the importance of data time stamps, development, and ML model background [3]. In addition to the challenges faced by software engineers, the organization also needs to adjust processes to suit the need for a constant feedback loop, experiment management, privacy and security management of the data, and control and use of the product responsibly, seeking fairness [18], [35], [37], [38].

III. Study Design

From our exploratory literature review, we mapped the ML product's life cycle phases (see Fig. 1): 1) problem definition and solution design; 2) product management; 3) data management; 4) model management; 5) software engineering; and 6) delivery/runtime. Based on these phases, we defined our search queries (described in Sections III-A1 and III-B1.)

Fig. 2 illustrates an overview of the study design. We followed the subsequent steps detailed in the next sections: grey literature review (GLR) and systematic literature review (SLR), also known as MLR, initial/open coding, selection of core categories, and focused coding. The outputs of these phases are the documents analyzed, the coding, the memos, the categories, and the resulting conceptual map. The goal is to provide researchers, practitioners, managers, and organizations with an aggregation of practices throughout the entire life cycle of ML software products.

In an MLR approach, a systematic analysis of diverse literature sources is conducted. Apart from academic peer-reviewed papers, inputs from GL sources, including blog articles, were incorporated to capture emerging practices in the ML product life cycle from both practitioners and researchers [39], [40], [41]. By employing MLR, we aim to gather a wide range of information and conduct an interdisciplinary analysis of practices and methods discussed in academic and practitioner literature. This approach allows us to capture insights and perspectives from various disciplines and stakeholders.

We adopted a systematic data analysis strategy. Our analysis employed two techniques from Charmaz's grounded theory (GT) variant [42], [43], which guided us in constructing a conceptual map via Initial and Focused Coding procedures. During Initial Coding, we examined the data word-by-word and line-by-line. Subsequently, we categorized the data based on the most frequent or relevant codes through Focused Coding. We did not employ all the procedures of GT in our method or propose a theory; we only utilized coding and focused coding techniques to support our research objectives. These techniques allow a systematic analysis of unstructured data from documents, guide discussions to decide which vocabulary we adopt when similar practices have different names, and provide traceability and reproducible results.

This section describes our MLR protocol, the selection and exclusion standardized criteria, and analysis procedures conducted to map the state of practice in ML product development.

A. Grey Literature Review

Our GLR workflow, as outlined in Fig. 3, has three phases: planning, execution, and data analysis. First, we planned our study by specifying the background and objectives described in the introduction.

1) Planning: The use of GLR is still a challenge due mainly to the need for proper guidelines for supporting GL in software engineering (SE) [17], [44]. Adams et al. [45] and Zhou [46]

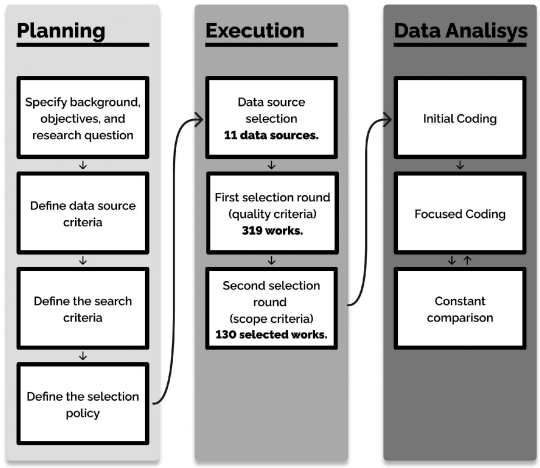

Fig. 3. Grey search flow was utilized in this work, involving three distinct phases. The first two phases, namely Planning and Execution, are integral parts of the literature review process, while the third phase focuses on data analysis.

categorize GL as *shades of grey* regarding outlet control and source expertise.

We defined the following gray scale: 1) Project Development Reports; 2) Foundation website articles and white papers; 3) Magazine and news website articles; 4) Community Wiki pages; and 5) Blog posts. Blogs are the recommended source to understand the current views of the practitioner's community [44], and we focus on this data source in the present study.

Our criteria for data sources aimed to select the most relevant from the practitioners' perspectives. These criteria were: 1) excluding data sources focused on technical issues, such as image processing and libraries for reinforcement learning; 2) excluding data sources that implicitly the purpose of the posts is to share the developments and information about a particular company; 3) including data sources that present information and posts that express learnings, techniques, methods, and challenges in developing an ML product; and 4) including data sources with posts organized using tags, search fields, or categories that facilitate access to relevant information for this research. We targeted websites based on their reputation, which is the most helpful criterion for choosing a GL source. Blogs have more volume data and are more challenging to control than other GL sources, but they provide the most up-to-date discussion of the practitioners. We address the quality assessment through standardized selection and exclusion criteria [44].

We grounded our search strategy on a query string tailored for GL data sources. Since GL data sources do not support advanced queries (e.g., enabling logical connectors), it is more helpful to manipulate simple and relaxed search strings [17]. Therefore, we defined seven simple query strings: 1) "ML product"; 2) "machine learning products"; 3) "ML problems"; 4) "ML data"; 5) "ML models"; 6) "ML and software engineering"; and 7) "ML deployment and runtime." It is important to note that each blog has particularities and different ways of presenting the posts. Therefore, we incorporated tags and filters related to our objective adapted for each blog in addition to the query strings. The Execution section describes the method of collection and its specifics.

Regarding quality criteria, the protocol for our first selection round was partly based on the guidelines suggested by Garousi et al. [39]. The selection criteria of this first round were as follows.

1) *Reputation:* We discarded works whose authors did not have experience in the area or other works published in the field. However, we considered works published by reputable organizations.
2) *Methodology:* We discarded works without clear objectives and methodology.
3) *Objectivity:* We discarded works with potential business interests; we considered only works with data-supported conclusions.

The protocol for our second selection round focused on the scope of the articles so that we could select only the essays pertinent to our objectives. Initially, we examined the titles, followed by the contents, and we favored the selection of articles that:

1) focus on the development of ML products;
2) discuss peculiarities and challenges for developing or managing ML products;
3) report the use of practices applied to ML products;
4) discuss the ML product life cycle, process, or workflow;

At the same time, we discarded posts that:

1) address some topic (e.g., ML in user experience) with ML products mentioned only for the background contextualization;
2) focus on a single ML tool without contextualizing it in the ML product life cycle.

2) Execution: We performed a Google search to identify the blogs most used by the community using the string ("ML blogs" OR "machine learning blogs"). This search return, for the most part, posts ranking and indicating the best blogs. We selected the top 5 to analyze and identify the most cited community and relevant blogs. They are the following:

1) Best Machine Learning Blogs to Follow (Towards AI);
2) Top 50 Machine Learning Blogs, Websites & Influencers (Feed spot);
3) 40 Must-Read AI/Machine Learning Blogs (Springboard Blog);
4) The Best Machine Learning Blogs and Resources (STX Next);
5) What Are The Best, Regularly Updated Machine Learning (Neptune).

After a first analysis and considering the bases indicated by practitioners, we selected 17 bases. Then, we conducted a more careful assessment, considering the criteria established in the planning, resulting in 11 data sources described in Table I.

In the following, we present an analysis of the data sources selected for this work. We divided the data source into three groups: academic blogs maintained by academic institutions, representing two data sources selected for this research. Also, we identified and selected blogs maintained by companies with informative posts, case studies, and technical documents. When selecting the posts of these blogs, we need more attention to

TABLE I
COMPREHENSIVE LIST OF THE DATA SOURCES ANALYZED AND SELECTED FOR INCLUSION IN THIS STUDY

Name	Type	Observations	Selected
ML.CMU	blog	More related to specific issues and models for ML do not talk about developing a product. posts are organized into two educational and research categories.	Yes
amazon.science	website	Contains an easy-to-use search field, and it is possible to use filters to select the search area. Contain posts related to technical reports, working papers, and product documentations.	Yes
Distill	blog	Does not have many posts and a search tool. We found peer-reviewed publications, commentary, and threads.	No
google AI	website	Has many and essential posts, but most of them pretend to share the information related to google. Contains information posts, product updates, researches, stories, and technical reports.	Yes
Neptune	blog	It is organized with tags to search big themes (ex.: ML Model Management and MLOps) and has a search field. It is possible find guides, articles, technical reports, and product documentations.	Yes
Bair berkeley	blog	Does not have a research tool and significant results for this research. Contain researches and peer-reviewed publications.	No
OpenAI	website	Does not have a research tool, and the discussion is more related to specific issues and information about OpenAI scholars, events, announcements, and researches.	No
Machine Learning (Theory)	blog	It focuses on technical discussions about experiments on models.	No
Deepmind	blog	It presents Arvix papers and has a search field, but it does not have tags that help select posts.	No
O'Reilly	website	Mostly books and videos, which are not being taken into account in the research.	No
Analytics Vidhya	website	The posts are organized by categories and according to the publication date. The search field is for the entire page, which makes it difficult to search for relevant information. We found use cases, technical discussions, technical guides, and researches.	Yes
Medium	blog	It has many readers and writers, search fields, and organizes posts with tags that help select. The blog contains use cases, technical reports, product updates, research, guides, and stories.	Yes
Towards DS	blog	It has a search field and tags that help in the selection of most related posts. We found use cases, technical reports, product updates, research, guides, and stories.	Yes
cisco	blog	The posts are organized by categories have a search field and tags. We found product updates, researches stories, and technical reports.	Yes
MIT	magazine	Posts are more focused on building the models and more technical details about the models. Contains technical research, peer-reviewed publications, and technical discussions.	No
Harvard Business Review	magazine	It has many essential posts for the theme of this study, but it has many private posts. We found peer-reviewed publications, and technical discussions, interviews, use cases, guides, stories, and reports.	Yes
Microsoft	blog	It has a search field, but most of the posts are related to presenting new technologies and developments of the company. We found technical reports, news, stories, interview, and events.	Yes

The table includes the name of each GL source, a brief description or observation, and whether or not it was chosen for inclusion in the current research.

identify posts with bias or seeking to promote the company. Altogether, we selected four data sources linked to the companies.

The third and last group was named Independent, as the practitioner's community maintains them. We selected all three data sources related to this group. In the following, we detail the data sources and how to collect the information in each one, organized by the groups.

Academics: We analyzed the first blog, ML.CMU, where students, postdocs, and faculty at Carnegie Mellon University (CMU) write the posts. During our research on the product at ML and CMU, we discovered that the posts are categorized into research and educational sections. The Educational category fits more into the objective of this research; therefore, filtering through it and following the selection criteria established in the planning, we collected seven posts related to the life cycle of the ML product. We also utilized the *Harvard Business Review* magazine in this study. As it is a broader magazine, we searched for "machine learning," which resulted in 260 results, 31 of which were selected to compose this study.

Companies: Next, we analyzed Amazon Science. In addition to the search field, research area, and date filters, the blog has selected "information and knowledge management" and "machine learning." We do not use the filter date to collect all information. Altogether, we received 72 results, of which we selected five. Most posts are related to more technical aspects of building models and presentations on Amazon. We also analyzed Google AI. Most posts are related to building models to solve specific problems (e.g., Playing a duet with a computer through ML). Others are related to publicizing the company's products and events (e.g., how ML in G Suite makes people more productive, Google Cloud announces Machine Learning startup Competition). On the other hand, we identified posts created in partnership with Google. The first was called Pair, created by a multidisciplinary team from Google that works with

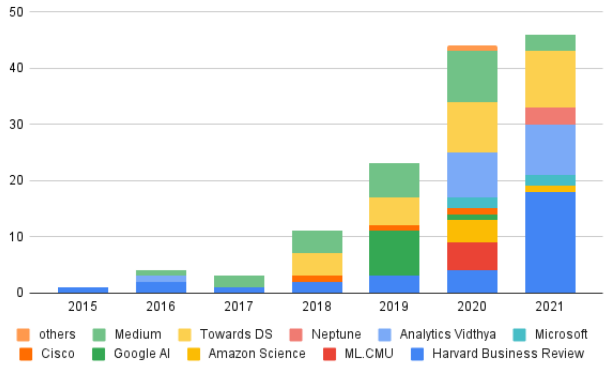

Fig. 4. Distribution of the number of selected posts per source, categorized by the year of publication.

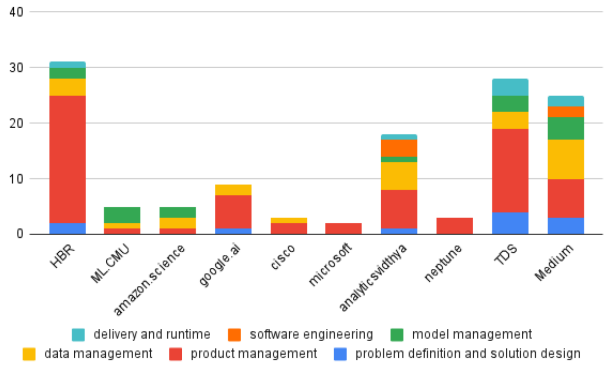

Fig. 5. Distribution of the number of selected posts per source, organized by categories.

research and product development. Other posts were collected from Google Developer and directly from Google AI, totaling nine posts. We also chose the Cisco blog because it has relevant posts related to product management and data management. With the help of tags, we collected a total of three posts.

Independents: The blogs Analytics Vidhya, Medium, and Towards Data Science were selected because they have posts related to this study. For many posts and a more complicated way to filter, the search was more manual in Analytic Vidhya, selecting the category ML or deep learning. Altogether, 18 posts were selected. Medium and Toward DS, on the other hand, have a vast number of posts, being the main ones responsible for the results of this research, which add up to 53 posts.

All 384 posts were initially collected, and after filtering the analysis of the works, taking into account the protocol and the relevance verification, we used 130 in this research.

Fig. 4 displays the distribution of the selected articles according to their publication year. We did not apply a starting date filter. The first selected publication is from 2015, indicating how recent our study subject is. We can also observe a growth in publications between 2018 and 2021, showing an increased interest in this topic.

We collected information such as the year of publication and the author's info during the execution and analysis. We also made observations to help define and justify the selected works. In addition, we also categorized the works into one of the six categories that guide this research.

We considered the theme of the posts to define the categories for each code and, then, the distribution of practices in each category. In addition to the general postanalysis, we considered the number of practices identified in each work. For example, Fanous's post [47] discusses data-related aspects and supports 13 methods mapped in the present research and eight related to the data management category.

Fig. 5 presents a bar chart with the distribution of selected works in each database divided by categories. We can identify some critical results. Most selected posts in *Harvard Business Review* and Toward Data Science relate to product management. At the same time, Medium has a significant amount of information related to data management. We also identified that these blogs are more related to understanding the problem and aligning the business. Therefore, we can infer that these three blogs are essential for research related to product engineering in ML. They have a large amount of information about the process, challenges, and techniques of the ML product life cycle.

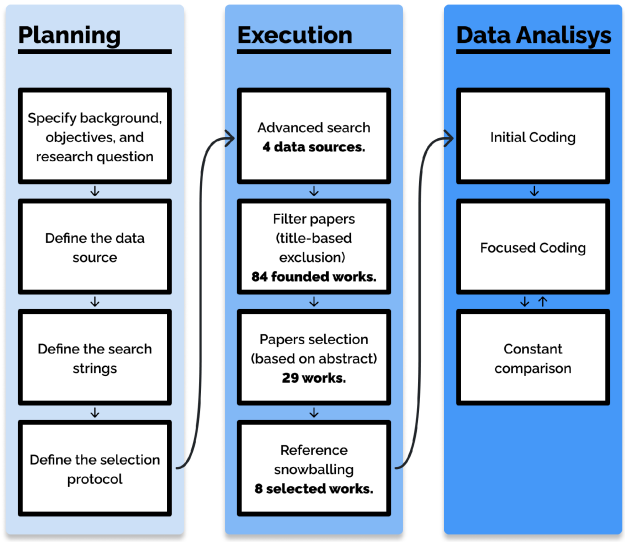

Fig. 6. Systematic search flow used in this work. This process is divided into three phases, with the first two (Planning and Execution) forming part of the SLR process and the third phase being data analysis.

B. Systematic Literature Review

As with the GLR, we performed the SLR in three phases: planning, execution, and analysis [48], as presented in Fig. 6. In the following, we detail each of these processes.

1) Planning: Our objective in executing the SLR is to identify works, terminology, practices, and techniques known by the academy [24]. We first analyze the information provided by academics and then compare that with the technical vocabulary and practices employed by practitioners.

The search sources were online digital databases, including IEEE Xplore, ScienceDirect, ACM Digital Library, and Springer Link. We verified Google Scholar but found almost the same publications on this database.

To construct our query string, we define the following keywords from the questions and the categories defined from related work: *machine learning*, *product management*, *data management*, *model management*, *software engineering*, and *practices*.

This research focuses on *machine learning*, and we know that ML is only a type of *AI*. However, it was necessary to insert these terms because some articles refer to AI as a synonym for ML. We noticed that some papers refer to *product engineering*, *product management*, and *product development* as synonymous. Therefore, we adjust our query string to accept all these terms.

We establish six initial queries after defining the keywords and identifying the synonymous. Then, aiming to collect the maximum amount of information and reach the saturation of relevant information, we performed tests on Google Scholar. It is a base that includes publications from the relevant databases used in this study. Table III (in Appendix A) presents a detailed list of search strings (without variations with synonyms) organized by categories. After the tests on Google Scholar, aiming to collect as much relevant information as possible, in addition to the search queries, we performed tests with more generic searches, such as "Machine learning products," "problems," and "life cycle." Altogether, 25 search strings were defined, containing the most generic to the most specific terms and connected with the categories defined from the related work and the background.

We define four steps to select the papers:

1) *advanced search in scientific repositories:* automated research on a scientific basis. It only included papers from journals, conferences, congresses, or symposiums;
2) *filter the papers:* a title-based exclusion following the exclusion criteria: a) "improper source"; b) "repeated"; c) "studies that are not written in English"; and d) "if the paper has less than three pages";
3) *paper selection:* based on the abstract and, in some cases, their content too. We also verified if it addresses our problem and research questions;
4) *reference snowballing:* to ensure that we could find as many studies as possible, including studies not retrieved by our search strings.

As also defined in our GLR planning, in the work selection phase, we include works that:

1) discuss practices used to develop an ML product;
2) present the challenges of applying ML in a company;
3) present the challenges and possible methods of understanding the problem and aligning it with ML development;
4) discuss peculiarities and challenges for developing or managing ML products;
5) report the use of practices applied to ML products;

Furthermore, we determined to exclude works that:

1) present as the main topic more technical discussions about the mathematical construction of the model;
2) present construction and technical details of tools used during the ML life cycle.

2) Execution: When we compare the execution of the SLR with the GLR, the SLR process is more straightforward, as it is not necessary to maintain a constant validation of the quality of the papers found in the scientific databases. In this way, our focus was to find the most relevant works linked to our research objective. We checked the first 100 results, sorted by relevance and most cited by papers.

We first analyzed the IEEE database, applying our queries in the advanced search that resulted in 12 selected works. When applying queries related to the product management category (a total of three variations), we identified that the results found were more generic. Thus, we selected papers related to product management and data management. Moving on to the data management category, we also defined the terms data pipeline, feature, data mining, and data science process, which resulted in new work. At the end of the searches, we found repeated documents with an average of six new works collected for each query. Analyzing the results obtained in the ACM Digital Library, we initially identified 14 papers related to the application to business, best practices, data management, model management, and MLOps.

The execution on the Springer Link database was performed differently than IEEE and ACM due to their different way of doing advanced searches. Therefore, in addition to looking for the search string, we use the filters Content type, Discipline, and Subdiscipline to select the works related to this search. We applied this strategy from the most straightforward searches to the most generic ones, as was done in the other databases selected in this study.

C. Data Analysis

Our coding process is divided into phases. First, we examine the data word-by-word highlighting excerpts linked to our research question (Initial Coding), deriving from their practices for ML product engineering. We constantly compared the coding from different articles and grouped common practices, even when different posts used different terminology. We build a codebook with code definition and excerpts from documents. Then, we select the most frequent codes and groups to help categorize the data (Focused Coding). Finally, we analyze the data and core categories to compare, understand the relationship, and integrate them into a cohesive conceptual map.

During the Coding process, we integrated the blog posts with the papers in the same process, separating them only to organize the references. We extracted quotations (segments of texts) for each document and assigned codes to these quotations. All this constant comparison happened in parallel with the analyses of documents from GL and papers from peer-reviewed literature. It helped identify the standards the community adopted and define the most appropriate terminology to represent the code. However, some terms vary among papers and posts, as shown in the tables in Appendix B.

We evaluated the resulting codebook and tested whether each code could replace the corresponding quotation without changing its meaning. It helped understand and refine the nomenclature of the different concepts (codes). However, it was inserted in a similar context (e.g., increase the computational capability or take advantage of the latest hardware [37], [49]).

In the following, we present some examples of extracted practices alongside the original excerpts associated with them:

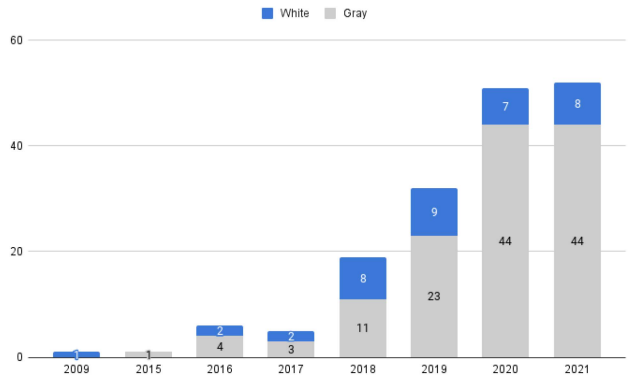

Fig. 7. Chronological distribution of the literature analyzed, broken down by year of publication. The graph illustrates the number of white literature and GL sources reviewed, visually representing the distribution of literature over time.

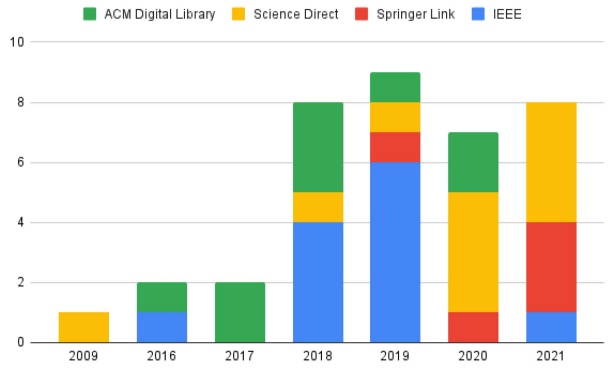

Fig. 8. Distribution of the number of selected papers, categorized by source and publication year.

1) *define the desired outcome:* "define the objective function (outcome) and metrics..." [50];
2) *review the literature:* "the review of related literature (RRL) step of the ML workflow involves reading up on existing approaches, datasets, and others resources" [51];
3) *data requirement:* "figuring out what data are needed for a specific product or feature is the first and most important step in scoping data requirements." [52]).

Our data are available in a replication package (as described in the Data Availability section at the end of this work), providing a detailed chain of evidence so that any researcher can verify all codes and the most relevant excerpts. In Appendix C, we provide a glossary that includes detailed descriptions of each practice in the conceptual map.

IV. Results

Our study involved gathering a total of 130 papers, comprising 37 sourced from peer-reviewed repositories and 93 sourced from a range of websites, blogs, and magazines. To provide an overview of the selected documents, we have included a visual representation of their distribution by year in Fig. 7.

We also analyzed the results by comparing the source of selected works with the year of publication. Fig. 8 shows the distribution of selected publications by year. As a result, we can observe that ScienceDirect published the oldest among the selected works, and that there was a significant increase in publications starting in 2018. In comparison to 2018 in Fig. 4, the same year in Fig. 8 presents almost the same number of selected posts and papers, followed by a significant increase in the publication of posts, while scientific papers did not follow the same pace, maintaining an average of ten papers per year on the subject. This is likely due to the process of peer review and quality assurance in academic publications. This high increase in GL posts is expected as there has been a huge volume of work in ML and data science in the tech industry since then.

The coding process resulted in 91 practices and seven methods related to the ML product development process (e.g., *verify how necessary ML is for the product* [53]). The Focused Coding resulted in six semantic domains or categories: (S1) problem definition and product design; (S2) product management; (S3) data management; (S4) model management; (S5) software engineering management; and (S6) delivery and runtime, as shown in Fig. 9. Our objective is to provide an extensive overview targeted at enhancing systematic knowledge available to managers. To achieve this, we conducted a multidisciplinary survey, exploring concepts, methods, and practices related to this topic. While Fig. 9 presents the main domain, this section discusses all the concepts involved. By building upon this information, managers can assess their knowledge and apply it to their development processes while also recognizing areas that require further improvement. This critical perspective enables them to better understand current practices and how they can contribute to their work. Appendix B presents the emerged methods and practices grouped by categories. In addition, Appendix C provides an overview of the methods and practices and their descriptions or definitions, and Appendix D presents the complete conceptual map.

To illustrate the process, Table II and Fig. 10 show the concepts related to the *problem definition and solution design* semantic domain/category. Table II presents each emerged practice/code, the list of documents containing the practice, and the source this practice is found (in SLR, GLR, or both). From Table II, we can infer the relevance of each practice by its code density (frequency in which it is cited in documents) and identify research opportunities (practices found only in GL). In the Focus coding, we group these practices based on quotation and definition/context similarities, producing the conceptual map of the *problem definition and solution design*, depicted in Fig. 10.

In the *problem definition and solution design* category (see Fig. 10), we have 16 practices related to requirement engineering, ML hypothesis formulation, and business improvement. For instance, *continuous validation business* practice assists the business growth and verifies if the product fulfills the established goals and requirements [50]. A practice frequently mentioned by practitioners is the *definition of the desired outcome*, as the costs and challenges of developing an ML product are high.

In the *product management* category (see Fig. 11), we mapped 19 practices associated with ML product management activities. The practices are organized into four subcategories: *compliance*, *management*, *improvement*, and *maintenance and evolution*. One of the most cited is *improvement using explicit and implicit*

Fig. 9. ML product management overall conceptual map.

TABLE II
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *PROBLEM DEFINITION AND SOLUTION DESIGN* CATEGORY

ID	Source	Practice
1	Papers/posts	Business understanding and validation [4], [5], [14], [23], [49], [50], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68]
2	Papers/posts	Identifying the right problem to solve [3], [23], [56], [66], [67], [69], [70], [71], [72], [73], [74], [75], [76], [77]
3	Papers/posts	Verify how necessary is ML for the product [3], [7], [23], [49], [50], [53], [63], [65], [68], [70], [78], [79], [80], [81], [82], [83], [84], [85], [86], [87], [88]
4	Papers/posts	Define the role of ML on product [7], [8], [49], [53], [56], [64], [65], [70], [82], [83], [86], [89], [90], [91], [92], [93], [94], [95]
5	Papers/posts	Statement of expectation and intention [4], [6], [7], [8], [18], [49], [50], [53], [57], [63], [64], [65], [70], [82], [83], [84], [92], [94], [96], [97]
6	Papers/posts	Establish what is the outcome and what the data can offer [6], [14], [55], [56], [57], [59] [49], [65], [67], [79], [82], [88], [98], [99]
7	Papers/posts	Set expectations for adaptation [4], [6], [94]
8	Papers/posts	Apply Design Thinking to understand the user deeper [5], [96], [100], [101]
9	Papers	Lean Canvas [5], [14], [102], [103]
10	Papers/posts	Apply prototyping to validate and align the objectives [6], [83], [100], [104], [105]
11	Papers/posts	Define the Desired Outcome [4], [7], [9], [50], [51], [51], [52], [57], [62], [63], [65], [66], [70], [80], [83], [84], [94], [104]
12	Posts	Translate user needs into data needs [57], [58], [71], [106], [107], [108], [109], [110]
13	Papers/posts	AI companies need to create an ontology [111], [112]
14	Papers/posts	Conduct preliminary experiments and track [8], [13], [65], [92], [104], [113], [114]
15	Papers	Include a goal description on requirements [113]
16	Posts	Reengineer some business processes [8], [49], [65], [92]

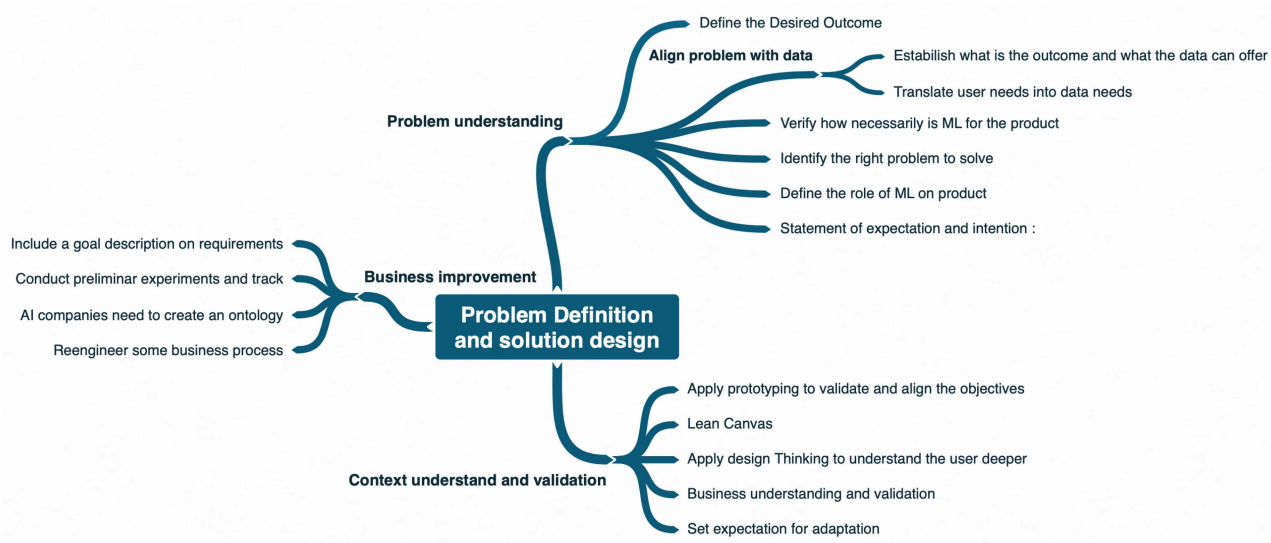

Fig. 10. Overview of the *product definition* category, including the key practices for *problem understanding*, *context understanding and validation*, and *business improvement*.

user feedback, which consists of promoting experiments and defining metrics to evaluate ML products with beta users and implicit feedback from end users [85], [115]. A widely cited practice is *multiple interactions with users and stakeholders*, so engineers can use their feedback to improve and assess the models [63], [82].

The *data management* category (see Fig. 12), with 19 practices, encompasses activities related to managing data used to train models, from *Ensure the reliability and availability of data* [50], *data collection and evolution* [83], [116], and *data cleaning* [80], [81] to *data labeling* [52], [84]. The practices are grouped into five subcategories: *data planning,*

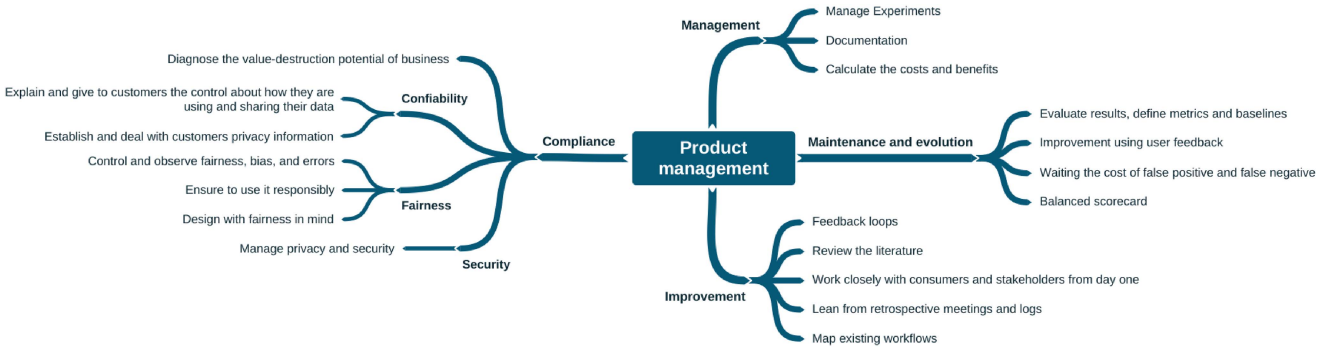

Fig. 11. Overview of the *product management* category, including the key practices for *data understanding*, *management*, *improvement*, *maintenance and evolution*, and *compliance*.

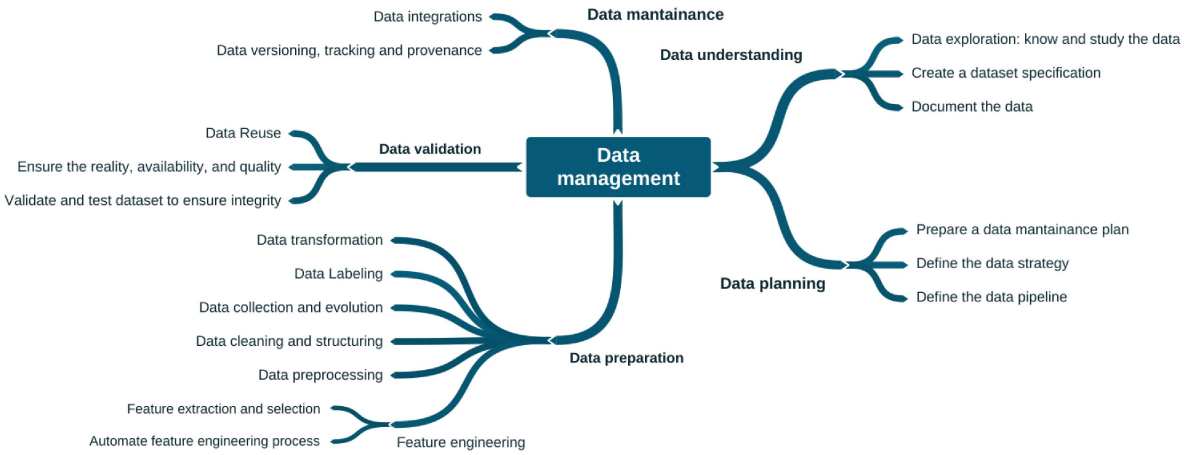

Fig. 12. Overview of the *data management* category, including the key practices for *data understanding*, *data planning*, *data preparation*, *data validation*, and *data maintenance*.

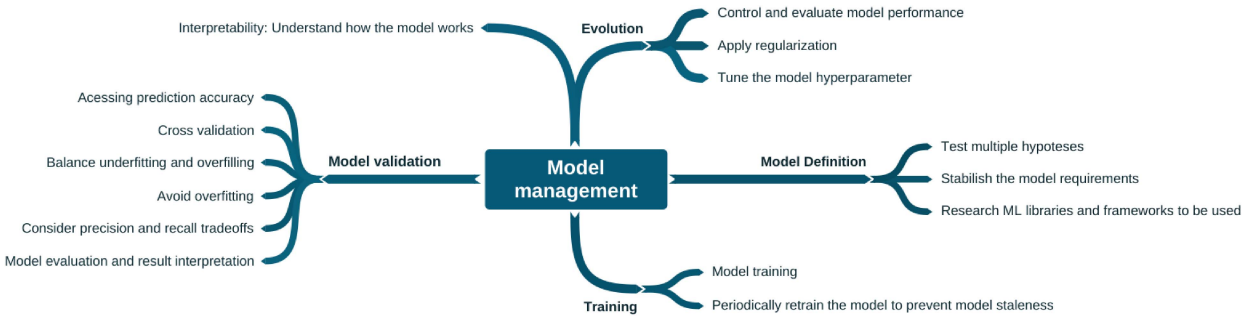

Fig. 13. Overview of the *model management* category, including the key practices for *model definition*, *training*, *model validation*, *evolution,* and *interpretability*.

data understanding, data preparation, data validation, and *data maintenance*. Data scientists, engineers, and managers explore the data to check and identify possible errors and understand if the data are necessary for the model hypothesis. Applying practices to ensure data quality (*data management* [90], *data reuse* [47], [117], and *data versioning* [118]) leverages the reuse of datasets across diverse products. Data engineers mostly perform the practices and methods of this category.

The *model management*category (see Fig. 13) consists of 16 practices to manage the development and continuous model training. The practices are organized into five subcategories: *model definition, training, model validation, evolution*, and *interpretability*. The practice *research ML libraries and frameworks to use* [79], [81] identifies the communities and maps the open sources; it then selects libraries based on licensing and uses terms that suit the *model requirements* [84]. The practices *feature engineering* [51], [80] and *model training* [90], [118] refer to all the activities performed to extract and select features, train, and tune the collected data. The *model evaluation* [51] practice consists of testing the output model against test datasets using predefined metrics. The practices and methods of this category

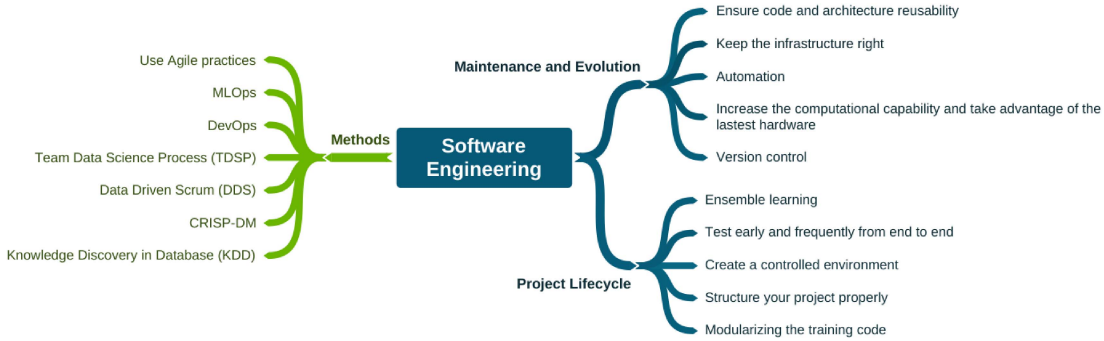

Fig. 14. Overview of the *software engineering* category, including the key practices for *methods*, *project life cycle*, and *maintenance and evolution*.

Fig. 15. Overview of the *delivery and runtime* category.

are performed mainly by data scientists, with the technical support of data engineers and software engineers.

The *software engineering* category (see Fig. 14) consists of ten practices and seven methods referring to source code management activities. The goal of the software engineering team in an ML product ranges from the product architecture definition to integrating the ML model with a software system. These practices guide managers and developers to build ML components and infrastructure. Practices such as *modularizing train code* [86] and *ensemble learning* [83] are used to isolate the trained models to ensure minimal dependence among them. These adjustments in software engineering practices have been discussed in the context of hidden technical debt and troubleshooting integrative AI [1], [15], [16], [28].

On the other hand, software engineering has more advanced methods that can also assist in the development of ML products. For example, adopting *DevOps* [119], [120] best practices is a prudent approach to help data scientists overcome common challenges in software production. However, unlike DevOps, *MLOps* [14], [121], [122] is more experimental in nature. Data scientists and ML engineers must tweak features such as hyperparameters and models while managing data and code bases for reproducible results. *MLOps* also encompasses practices for collaboration and communication between data scientists and operations professionals.

Besides MLOps, experts have developed several methods, based on agile methodologies, that describe the data science life cycle, such as *CRISP-DM* [3], [123], *DDS* [3], and *Knowledge Discovery in Database (KDD)* [3], [58]. Microsoft launched another popular method in 2016, called the *TDSP* [14]. TDSP combines Scrum and CRISP-DM, making it an agile iterative data science method that delivers predictive analytics solutions and ML applications.

With ten practices, the *delivery and runtime* category (see Fig. 15) covers the problem of managing the infrastructure, deployment, and monitoring of ML products. It encompasses *Deploy and operationalize the model* [51], [124] and *Build specialized and reusable pipelines* [84], [90], which involves building pipelines for the company's specific context to facilitate the loading and editing of data and experimentation with different algorithm permutations [79].

Another technique widely used in this phase is *continuous model monitoring* [56], [125]. It ensures that ML models deployed in production operate optimally. Understanding how quickly model performance degrades over time helps prioritize monitoring efforts. If the model's quality drops significantly in a day without updates, an engineer must monitor it continuously. *Continuous model monitoring* is necessary to maintain the scalability and robustness of deployed models and to refresh models when needed based on changes in summary statistics of the data. Real-world data are subject to various changes, and continuous monitoring identifies potential errors in production. Therefore, monitoring ML components in production for optimal performance and reliability is essential [3], [18], [56], [58], [79], [90], [125].

Finally, one of the most critical practices in ML is to *deploy and operationalize the model* [69], [111], [125] actively. This involves constantly feeding new data to deployed models to enable them to adjust to real-world changes, such as new categories and levels. Integrating predictive models into software products and systems involves invoking the appropriate model at the correct point. Deploying a model is just the beginning,

as it often requires retraining and performance checks [125]. To ensure continuous learning, ML models should undergo training with new data, and a versioning system that handles model parameters, configuration, feature pipeline, training, and validation datasets must be in place. After establishing well-performing models, they can be operationalized for consumption by other applications. Predictions can be made based on business requirements, either in real time or on a batch basis. The prevailing method for deployment is to expose models through an open application programming interface (API) [23].

V. Discussion

Our results indicate that managing ML products is extensive and spans many factors and themes, which are not all currently mapped or studied. Furthermore, there are unsolved challenges that managers, researchers, and practitioners should be aware of. In this section, we discuss the results.

A. Impact of Problem Understanding

Several organizations have started to adopt data-driven software projects and generate more prescriptive insights. They aim to make decisions about their business plan, operations, products, and services. *Identify the right problem to solve* [70], [71], *Verify how necessary ML is for the product* [3], *Statement of Expectation and Intention* [50], [53], and *Define the role of ML on the product* [89] are some practices core before adopting ML components. In traditional software projects, we often witness similar practices when defining features. However, when it comes to ML, these practices become more time consuming and are typically carried out in an intensely iterative manner [2]. These practices help to correctly establish the role of ML in the product or organization and how it will impact the current process.

The practice of *Include a goal description on requirements* [113] and *Define the desired outcome* [57], [104] is employed by academia and practitioners. These methods contribute to aligning goals, validating ML models, and ensuring their effectiveness. They clarify expected outcomes, facilitate appropriate algorithm selection, and align the models with desired business objectives. Defining the desired outcome enables better alignment between the problem and the model, optimizing algorithm selection for goal achievement. It is essential for setting clear objectives, guiding decision making, and enhancing the relevance and impact of ML models.

The practices listed in the *problem definition and solution design* and *product management* categories are fundamental for model validation. We observed studies that demonstrate this concern and mention practices such as *Establish what the outcome and what the data can offer* [49] and *Define the desired outcome* [70]. The alignment of objectives impacts the management and testing of the dataset *Create the dataset specification* [71] and *Ensure the reliability, availability, and quality of data* [108] and the model after training *Establish the model requirements* [126] and *Model Evaluation and result interpretation* [91], [124].

B. Regulations, User Agreement, and Data Privacy

ML products negatively impact organizations, users, and society if misused. Therefore, organizations interested in starting development must *Ensure to use it responsibly* [93], as they may be dealing with personal and sensitive data. *Diagnose the value-destruction potential of a business* [54], even before starting the development, to verify and analyze if the identified risks can negatively impact the organization's values.

Establish and deal with customer privacy information is a practice often cited by practitioners when ethical issues regarding the use and sharing of this data are essential to agree with users to ensure the security of these data. Governments created data protection law regulations such as the EU General Data Protection Regulation to ensure users' data privacy. These regulations help prevent ML from overriding citizens' rights. Organizations and regulations should consider: making it clear and available whenever necessary to the user the reason for using their data, the relevant costs and benefits of sharing, and notifications about data breaches [109]. Practices such as *Explain and give customers control about how they are using and sharing their data* [127] are essential to ensure alignment and transparency in the use of data.

C. Team Structure and Engineers Skills

The results suggest that practitioners are currently discussing challenges regarding data maintenance and quality practices. The most cited practices were: *data strategy*; *data collection and evolution*; *ensure the reliability and availability of data*; and *data cleaning and labeling*. The focus on the *data management* category is highly relevant, suggesting that the crucial concerns for engineers shifted from source code to data.

Therefore, ML product teams require more skilled software engineers and a shift from DevOps to MLOps. Versioning data and data schemes, training models, cleaning, reusing, labeling, and configuring automated pipelines are examples of software engineers' assignments in a project with ML modules.

D. Product and Project Management

The *product management* category presents more practices and methods than other categories. It suggests that practitioners face challenges adapting standard software engineering practices and workflows to ML product development. Managing the ML product workflow requires defining potentially new team roles, adjusting agile practices to incorporate ML design's experimental nature, and incorporating ML workflows, tools, and environments [4]. *Feedback loop* practice illustrates how this is an emerging research topic. In [16], one of the first papers on ML systems, the term *feedback loop* is a technical debt of ML systems when the model may directly influence the selection of its future training data or indirectly influence the training data of another model. However, in our coding, *feedback loop* appeared as an agile practice intensified over developing an ML system since the process is more experimental than standard software.

VI. Implications

In this section, we present the research implications as a practical guide to help professionals adapt to the significant impacts of ML in their fields. Our contribution is to discuss the implications of ML adoption for each perspective individually: engineers, managers, and researchers. Readers will find guidance on the issues they will likely confront, core concerns they should consider, and potential solutions the community has adopted. In addition, we outline ML-related topics that the academic community could explore in the future. By presenting these implications, we raise relevant considerations and prepare the ground for our discussion of unresolved challenges in Section VII.

A. Implications for Managers

Drawing from the reviewed literature, we identified several implications for managers when confronting the phenomenon of ML. These include the need for specific management and cultural paradigms, training individuals, structuring and evaluating the process of ML adoption, as well as anticipating the expected outcomes.

Managers, executives, and business strategists must collaborate closely with data engineers and scientists to develop the ML product strategy (see Table II). Managers should have a basic understanding of ML algorithms and data analytics to identify the problem types the available data can address, including prediction, recommendation, anomaly detection, and more. Obtaining training data presents a significant challenge for the viability of any ML module, often necessitating preprocessing, extract, transform, and load (ETL), and importing/scraping from external sources [12], [128].

In addition, the integration of ML requires specific changes at the organizational level, such as adjustments in team structure, roles, and processes. There are a few key considerations to make: 1) data scientists are responsible for conducting tests and experiments to validate the data; 2) it is crucial to enhance the collaboration between business and operational teams and one must continuously assess whether adjustments or the inclusion of new data is necessary to accurately represent the context and align with the business objectives and requirements; 3) the organizational culture must embrace a constant feedback loop [56], [58], vital for validating experiments and fostering collaboration across the entire team; and 4) managers also carry a heightened level of responsibility, particularly when working with confidential data and results as incorrect utilization of such information can have detrimental effects on the company.

From a process perspective, managers must adapt agile practices and data-centric processes to encompass the data processing, experimentation, and model testing stages. Team structures must also evolve to accommodate new roles, such as data scientists and data engineers. Cultural change becomes imperative once again. Both top-level and low-level management are responsible for fostering an environment where failure is allowed, and continuous improvement is sought. Close collaboration among data engineers, operations, software engineers, and data scientists must be fostered, and a well-defined collaboration flow for artifacts, communication, and hand-off should be established [129].

By considering these implications, managers can navigate the deployment of ML effectively within their organization while ensuring positive outcomes.

B. Implications for Engineers

ML models affect how engineers architect systems, interact with their peers, and add process activities, such as model debugging. From an engineering viewpoint, there is a need for model/data storage, versioning, and querying, which are closely intertwined. Model diagnosis assists ML developers in interpreting why the training process fails to achieve satisfactory performance and helps them identify the root cause within the model, data, or source code [128]. Model deployment pipeline and serving infrastructure configuration are challenging.

In addition to automating the code deployment pipeline, software engineers also need to automate the deployment of data and trained models. Consequently, tool builders must focus on developing automation tools that enhance explainability and enable actionable insights into these processes. In contrast to non-ML systems, requirements for ML systems usually involve many preliminary experiments, and it demands that requirement engineers have a strong technical background in ML and data analysis. Therefore, ML development requires a broader range of skills beyond programming abilities [2].

Data engineers possess skills in acquiring, cleaning, exploring, and modeling data. The substantial volume of data in ML development introduces new challenges for data engineers since personal computers do not support processing this volume. It imposes adaptations to the data science workflow, conducting experiments in cloud environments, potentially requiring environments with different configurations for each experiment. Hence, automating the provisioning of these environments with correct data access (typically, each data scientist has access to limited datasets) becomes crucial for the experiment success. Other automation may be necessary, such as data pipeline and preprocessing data customization. Consequently, data engineers have more attributions and perform a crucial role throughout the product life cycle. Additional skills are expected of data engineers, like programming, automation configuration, cloud provisioning, and feature engineering.

C. Implications for Data Scientists

Data scientists are responsible for dealing mostly with the "model management" category but also with the "data management" category. Automation is required to expedite the process of "testing multiple hypotheses." Practices such as "Automating the feature engineering process," "Defining the Data Pipeline," and "Researching ML Libraries and frameworks to be used" require a solid knowledge of software engineering principles and the challenges associated with deploying ML in a production environment and executing, versioning, and documenting experiments in the cloud. Tools such as metaflow[2] expand the

[2]https://metaflow.org

data scientists' attributions once they must develop, deploy, and operate various data-intensive applications and environments.

D. Implications for Educators

ML education often prioritizes algorithms and techniques and their application in controlled environments like specific datasets and Jupyter notebooks. ML education should also cater to requirement engineers and managers, emphasizing the fundamental concepts of effectively leveraging their organization's data. In addition, it would be valuable to incorporate realistic case studies that illustrate the nature of data, their potential applications, and the precautions to be taken when working with them. Thus, there is a clear need to address the integration of the ML workflow into the software product process, and this should be thoroughly covered by ML systems education in the future.

VII. Research Opportunities

Our MLR uncovered a range of significant challenges associated with managing and developing ML products. To identify potential areas for further research, we specifically concentrated on concepts mentioned exclusively in the GL. We conducted a follow-up investigation to broaden our search scope, using query strings related to these concepts, focusing on the sources identified in our SLR. The goal was to uncover more recent research or papers that might not have been captured in our initial search. In the following sections, we discuss the primary research opportunities we discovered, complementing the implications detailed earlier.

A. Multidisciplinary Research

From a research perspective, product engineering and software engineering are different areas of knowledge. While administration and business researchers focus on identifying business/organizational opportunities to develop ML products, software engineering researchers focus on technological challenges and development processes. Consequently, in terms of research and publications, these two domains tend to be separated. The variety of sources and search strings (see Tables I and III) evidence such observation. However, when analyzing the content available on the blogs, there is an evident intersection in practice, and we identified that many posts associate concepts from product engineering to ML module technical topics [65], [87]. Practitioners already work with a multidisciplinary perspective, where engineers, data scientists, and software developers are exposed to concepts from categories such as "product management" and "problem definition and solution design." Moreover, blog posts on management often introduce fundamental concepts of ML algorithms and software product development processes.

We conducted searches on 17 different data sources in the GLR (see Table I). We identified seven search query categories for the SLR (see Table III). The presence of diverse data sources and search queries leads to fragmented discussions and hampers a holistic view of the challenges in managing ML products. Multidisciplinary research is essential to better comprehend the challenges and opportunities associated with adopting ML modules. By encouraging cross-disciplinary dialogue and making the topic more accessible to a wider audience, multidisciplinary research can bridge the gap between technical and nontechnical perspectives, leading to more informed decision making in developing and deploying ML products.

Rather than solely focusing on specific aspects of managing ML products, such as risk management, fairness, bias, errors, privacy, or security, researchers should engage in multidisciplinary research that encompasses a broader scope. This approach considers the technical impacts of managerial decisions and the social–technical aspects of managing ML products. By adopting a multidisciplinary approach, researchers can explore the intricate interplay between technical, managerial, and social factors.

B. How to Adopt ML Systematically

Adopting ML products presents challenges and opportunities for businesses [5]. The GL offers a variety of case studies that provide insights into best practices for successful adoption [130], [131]. These case studies explore the gradual adoption of ML products, starting from proof of concepts to less critical and eventually critical modules. They emphasize the importance of experimentation, training, and adapting team structures throughout this process. However, research papers addressing case studies and best practices for adopting ML products are relatively rare [132]. One notable practice mentioned, which also presents a promising research opportunity, is reengineering business processes to align them with the capabilities offered by ML. This practice aims to ensure the optimal integration of ML technologies within the existing business processes.

Our study also cataloged a set of practices and methods associated with building ML products from the practitioners' viewpoint. While multiple blog posts cite most of the *data management* category practices, we found few industry-oriented texts on *agile practices* (see Table VII). We also found few publications in high-impact journals and conferences on using agile methods in ML systems development [132], indicating that it is an open research topic since agile is about dealing with uncertainties. In the case of ML products, the uncertainty is even higher. It suggests that adopting agile practices in ML workflows is an open topic for reflection by practitioners and researchers.

C. How to Assess the Quality of ML Product Practices in Organizations

The decision-making process of managers when allocating development and testing resources is often based on their past experience and intuition regarding the complexity of the new project compared to previous ones. However, this approach can result in inefficient resource allocation with building and maintaining software systems with ML modules when there is no previous experience in the organization. An organization can only achieve a competitive advantage when it has fast feedback loops and clear metrics [125].

Managers can consider several criteria and metrics to identify problems that would benefit from ML. They should *"verify*

the necessity of machine learning for the product" and define its role in addressing the problem. *"Establishing the desired outcome"* and *"understanding the insights that can be derived from data"* are essential. In addition, *"setting realistic expectations for adaptation"* and *"translating user needs into data requirements"* are important factors. However, an open question remains regarding whether the collected data are sufficient in quantity and quality. Ensuring that the right data have been collected in adequate quantities is crucial for the effectiveness and reliability of ML models [113]. While substantial research is available on the technical aspects of data and ML metrics [12], the field of management research concerning this topic remains open. A promising avenue for further investigation lies in developing strategies to effectively *"Evaluate Results, Define Metrics and Baselines."*

D. How to Qualify Managers for ML Practice

In managing innovative products, managers must possess self-learning skills to effectively address daily challenges due to the constantly evolving nature of knowledge in this area. A frequent question posed by managers is how to evaluate and mitigate the risks posed by adopting nondeterministic software and how to run these projects more effectively [37]. Uncertainty permeates various aspects of ML system development, including the data as a crucial component of the requirements and the inherent randomness of ML algorithms [2]. Such choices must be grounded in a thorough conceptual analysis of personal and organizational demands and perspectives. The Internet is a valuable resource for them, offering information through online forums and social networks. Social interactions on these platforms can give managers invaluable insights and ideas for their day-to-day work.

Our conceptual maps (see Fig. 9) and concept tables (see Tables II–VIII) may guide managers in prioritizing, improving their professional skills. However, a research opportunity is to elaborate on effective managerial procedures and case studies focused on ML product management. Doing so requires experience in market dynamics and thoughtful analysis of the potential value of specific ML modules and their embedded products and services. Lessons learned from case studies and guidelines for adopting ML products would be very valuable for managers in their work.

Fairness, privacy, and security are pivotal considerations in managing ML products. Key activities, such as "*Establish and deal with customer privacy information*," "*Control and observe the fairness, bias, and errors*," and "*Explain and give to customers control about how they are using and sharing their data*" are crucial from the inception of an ML project. Bridging the gap between technical research publications and the practical implications of product management in these areas presents an exciting research opportunity.

VIII. Related Work

In this section, we discuss and compare to our findings several related works from the academic literature and one from the GL. Previous studies [6], [19], [20] have primarily focused on identifying and addressing issues related to one or several phases of the ML systems development process (as depicted in Fig. 1). Serban et al. [18] performed an MLR and mapped 29 engineering best practices for ML systems categorized in the CRISP-DM phases [14], [27]: data, training, coding, deployment, teams, and governance. Based on their experience (not relying on data), they added six practices from software engineering. Through a survey, this work validated and analyzed the most used practices. Our research mapped some equivalent practices from [18], but we present a more refined set of practices, including a category *product management*, not covered in [18]. In addition, we analyzed the data systematically, generating the codes (practices) and categories from the data, allowing the repeatability of our research results.

Amershi et al. [1] conducted a case study with ML product teams at Microsoft to map their practices and challenges in building applications with ML modules. They present and discuss global practices, like *"End-to-end pipeline support"* and *"Model Evolution, Evaluation, and Deployment."* In contrast, our MLR maps the practices from academic and practitioner perspectives from various organizations and experience levels. The practices that emerged from the GL are more numerous and specific than the ones presented by Amershi et al. [1]. For example, the practice, *"Model evolution, evaluation, and deployment"* [1], would encompass the practices within the categories *"Model Management," "Software Management,"* and *"Delivery and Runtime"* of our GLR. The practices within the category *Problem Definition and Solution Design* of our GLR was not covered by Amershi et al. since they focus on the development process. As we did, they also identified data discovery and management as the most challenging practices in the ML product life cycle.

Martínez-Fernández et al. [133] executed a systematic mapping study considering 248 studies published between 2010 and 2020. They identified multiple SE approaches for AI-based systems classified according to the SWEBOK areas. This study analyzes academic works and discusses trends and aspects studied by academia. On the other hand, we bring a more profound discussion with data from academia and practitioners to discuss the practices used during the life cycle of the ML product.

Radanliev et al. [134], [135] have published works on integrating AI and Internet of Things (IoT) technologies into cyber-physical systems in the Industry 4.0. They discuss the potential benefits and challenges, such as increased efficiency, improved decision making, and enhanced security risks. However, there is a lack of focus on the management practices for ML products, which is the central focus of our study. Our work complements theirs by providing a practical perspective on managing ML products effectively. Another reason why we do not cover all practices of ML and IoT, or cyber-risk [136], for instance, is that we found them only in themed group discussions, not covered in our search queries. As demonstrated by the research opportunity highlighted in Section VII, this approach hinders a holistic understanding for managers.

From GL, Bourke [137] presented in July 2020 a video with a mind map with the main concepts in ML. In addition, Bourke presented reading suggestions and tools to assist in development,

TABLE III
SEARCH QUERIES USED IN THIS RESEARCH

Categories	Queries
Problem definition and solution design	(“machine learning” OR “ML Product” OR “deep learning” OR “Artificial Intelligence” OR AI) AND (requirements OR “problem definition*”) AND (methods OR practices) AND (”product engineering” OR “software engineering”)
Product management	(“machine learning” OR “ML Product” OR “deep learning” OR “Artificial Intelligence” OR AI) AND (“product management”) AND (methods OR practices) AND (“product engineering” OR “software engineering”)
Data management	(“machine learning” OR “ML Product” OR “deep learning” OR “Artificial Intelligence” OR AI) AND (“data management” OR “data pipeline”) AND (methods OR practices) AND (“product engineering” OR “software engineering”)
Model management	(“machine learning” OR “ML Product” OR “deep learning” OR “Artificial Intelligence” OR AI) AND develop* AND (“model management” OR “model pipeline”) AND (methods OR practices) AND (“product engineering” OR “software engineering”)
Software engineering	(“machine learning” OR “ML Product” “deep learning” OR “Artificial Intelligence” OR AI) AND (“solution design”) AND (methods OR practices) AND (“product engineering” OR “software engineering”)
Delivery/Runtime	(“machine learning” OR “ML Product” ”deep learning” OR “artificial Intelligence” OR AI) AND (“delivery” OR “runtime”) AND (methods OR practices) AND (“product engineering” OR “software engineering”)
General	(“machine learning” OR “ML Product”) AND develop* AND ((“problem definition” OR “solution design” OR “requirements”) OR (“project management”) OR (“data management” OR “data pipeline”) OR (“model management” OR ”model pipeline”) OR (“delivery” OR “runtime”)) AND (methods OR practices) AND (“product engineering” OR “software engineering”)

The table includes the basic search terms and phrases used to effectively search for and retrieve relevant works for each category.

TABLE IV
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *PRODUCT MANAGEMENT* CATEGORY

ID	Source	Practices
17	Papers/posts	Improvement Using User Feedback [6], [52], [53], [82], [85], [90], [115], [125], [138]
18	Papers/posts	Working closely with consumers and stakeholders from day 1 [1], [4], [63], [65], [66], [82], [104], [138], [139]
19	Papers/posts	Learn From Retrospective Meetings and Logs [4], [6], [9], [67], [79], [90]
20	Papers/posts	Review the Literature [51], [63], [69], [75], [82], [106]
21	Posts	Map existing workflows [70]
22	Papers/posts	Documentation [58], [86], [118]
23	Posts	Balanced Scorecard [62]
24	Papers/posts	Calculate the cost and benefits [54], [58], [60], [65]
25	Papers/posts	Feedback Loops [37], [56], [58], [64], [65], [66], [81], [83], [84], [85], [90], [117], [125], [140], [141]
26	Papers/posts	Manage the experiments [37], [142]
27	Papers/posts	A/B testing or Split Testing [6], [11], [51], [69], [82], [86], [115], [119], [143], [144]
28	Papers/posts	Weighing the cost of false positives and false negatives [3], [70]
28	Papers/posts	Evaluate Results, Define Metrics and Baselines [4], [11], [50], [51], [56], [63], [69], [75], [79], [80], [83], [84], [86], [87], [90], [107], [145], [146], [147], [148]
30	Posts	Ensure to use it responsibly [70], [93], [97], [109], [149]
31	Posts	Establish and deal with customer privacy information [109], [127], [131], [149], [150], [151]
32	Papers/posts	Design with the fairness in mind [18], [65], [70], [71], [72], [89], [109], [149], [152], [153], [154], [155], [156]
33	Papers	Diagnose the value-destruction potential of a business [54], [65], [70]
34	Papers/posts	Control and observe the fairness, bias and errors [6], [8], [51], [62], [65], [85], [92], [93], [94], [98], [110], [157], [158]
35	Posts	Explain and give to customers the control about how they are using and sharing their data [109], [127], [131]
36	Papers/posts	Manage privacy and security [8], [18], [37], [69], [71], [109], [151], [159], [160]

delivery, and runtime. He did not discuss the concepts and the implications of an actual ML component development. Nevertheless, Bourke presents and gathers the necessary vocabulary and the roadmap to help practitioners develop and understand ML systems. Differently, our research analyzes the documents and discussions from the perspectives of practitioners, managers, and researchers.

IX. LIMITATIONS

The choice of core and relevant documents may suffer from subjective bias. To reduce this bias, at least two authors conducted the selection process and the classification of papers. Senior researchers also oversaw the whole process. With considerable work on managing ML, mainly from GL, it would not

TABLE V
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *DATA MANAGEMENT* CATEGORY

ID	Source	Practices
37	Papers/posts	Create a dataset specification [57], [58], [71], [161]
38	Posts	Document the data [71], [109], [162]
39	Papers/posts	Data exploration: Know and study the data [14], [23], [58], [61], [106], [107], [125], [161], [163], [164], [165], [166], [167]
40	Papers/posts	Define the Data Strategy [8], [50], [52], [52], [57], [62], [71], [79], [83], [113]
41	Papers/posts	Define the Data Pipeline [8], [23], [47], [50], [57], [58], [75], [106], [113], [118], [124], [126], [162], [168]
42	Posts	Prepare a data maintenance plan [71]
43	Papers/posts	Ensure the Reliability, Availability, and Quality of Data [8], [18], [47], [49], [50], [52], [58], [59], [65], [79], [83], [84], [86], [108], [113], [126], [159], [169], [170]
44	Papers/posts	Validate and test dataset to ensure integrity [3], [18], [37], [55], [75], [104], [113], [125], [162], [171], [172]
45	Papers/posts	Data collection and evolution [8], [14], [23], [47], [51], [52], [56], [58], [62], [64], [65], [69], [71], [74], [75], [81], [83], [84], [85], [90], [91], [105], [108], [110], [111], [113], [114], [116], [118], [120], [124], [125], [159], [163], [165], [168], [170], [173], [174], [175], [176], [177]
46	Papers/posts	Data cleaning and structuring [18], [47], [49], [56], [58], [69], [72], [79], [80], [81], [83], [84], [86], [90], [105], [106], [107], [108], [110], [163], [174], [175], [176], [178], [179], [180]
47	Papers/posts	Data preprocessing [74], [125], [166]
48	Papers/posts	Data Labeling [3], [18], [47], [51], [52], [71], [81], [84], [159], [181], [182]
49	Papers/posts	Data Integration [18], [47], [104], [109], [124], [159], [183]
50	Papers/posts	Data versioning, tracking, and provenance [1], [4], [47], [64], [86], [90], [106], [111], [116], [118], [167]
51	Papers/posts	Data transformation [8], [13], [47], [111], [124], [163], [184]
52	Posts	Data reuse [47], [117], [167]
53	Papers	Feature extraction and extraction [55], [111], [178], [179], [185]
54	Papers/posts	Feature engineering [1], [3], [4], [18], [23], [51], [56], [72], [80], [84], [90], [107], [110], [113], [116], [120], [141], [159], [162], [168], [174], [177], [178], [180], [185], [186], [187]
55	Papers/posts	Automate feature engineering process [18], [188]

TABLE VI
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *MODEL MANAGEMENT* CATEGORY

ID	Source	Practices
56	Posts	Research ML Libraries and frameworks to be used [47], [75], [79], [81], [83], [84], [96], [168]
57	Papers/posts	Establish the model requirements [3], [81], [84], [126]
58	Papers/posts	Test multiple hypotheses [69], [83], [86]
59	Papers/posts	Model training [3], [23], [51], [56], [64], [72], [75], [83], [84], [86], [90], [91], [107], [111], [113], [116], [118], [120], [124], [159], [163], [164], [176], [182]
60	Papers/posts	Periodically retrain the model to prevent model staleness [75], [104], [144]
61	Papers/posts	Balance under fitting and overfitting [36], [71], [75], [107], [125], [169], [185], [189]
62	Papers/posts	Interpretability: Understand how the model works [8], [18], [36], [93], [127], [159], [162], [190], [191]
63	Papers/posts	Avoid overfitting [125], [172], [179], [192]
64	Papers/posts	Cross validation [179], [185]
65	Papers/posts	Apply regularization [178], [179], [192]
66	Papers/posts	Consider precision and recall tradeoffs [36], [60], [70], [75], [181], [185]
67	Papers/posts	Model validation [50], [55], [81], [83], [96], [114], [118]
68	Papers/posts	Model evaluation and result interpretation [36], [51], [52], [55], [56], [58], [64], [69], [72], [74], [75], [84], [86], [90], [91], [105], [113], [117], [118], [120], [124], [159], [163], [164], [165], [173], [176], [180], [193]
69	Posts	Assessing prediction accuracy [8], [49], [70], [99], [188], [192]
70	Papers	Control and evaluate model performance [11], [51], [107], [111], [113], [182]
71	Papers/posts	Tune the models and hyper parameters [13], [18], [75], [107], [110], [126], [178], [185]

be feasible to cover and condense all of them in this research. Therefore, we do not select books and academic workshops. However, by focusing on papers from journals and conferences and selecting the primary sources of practitioners, in addition to applying to snowball, we aimed to cover the primary literature on the subject.

Another issue is applying the search string in a website search engine. Many of these tools cannot address diversity and lack of text structure. We address this problem by adding tags to the query string, obtaining a more relaxed search string. The drawback of using a broad search string is the lack of more technical practices. For example, academics and practitioners have long discussed ML products' technical debts and the problem of component entanglement. However, we did not map practices that deal with this specific issue and other technical challenges practitioners face. To cover most of the practices and methods used in each category, we need to add to our GLR procedure a stage to define a flexible set of search terms, similar to Wen et al. [17], and potentially a few terms per category.

X. CONCLUSION

Organizations and manufacturers must evaluate their readiness and ability to undertake AI projects. This article explained the essential elements of the field and recommended which practices, roles, and profiles to develop within an organization, improving its chances of success in implementing ML products. We explicitly revealed the necessity of training managers, executives, and decision makers to have data analytics skills and basic knowledge of ML algorithms to make informed decisions on which approach to use and when. Blogs recommend exploring the gradual deployment of ML products, from proof of concepts to less critical implementation, before eventually incorporating critical modules. This article, specifically the

TABLE VII
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *SOFTWARE ENGINEERING* CATEGORY

ID	Source	Practices
72	Posts	Test Early and frequently from end to end [50], [80], [83], [84]
73	Papers/posts	Ensure code and architecture reusability [3], [13], [47], [51], [75], [79], [118], [194]
74	Papers/posts	Modularize train code [36], [86], [178]
75	Papers/posts	Version control [18], [47], [75], [86], [111], [114], [121], [126], [191]
76	Papers/posts	Ensemble learning [83], [86], [178]
77	Papers	Create a controlled environment [37], [111], [195]
78	Papers/posts	Increase the computational capability and take advantage of the latest hardware [37], [49], [75]
79	Papers/posts	Automation [18], [51], [90], [96], [114], [117], [121], [168], [191], [195]
80	Posts/posts	Keep the infrastructure right [87], [159], [176], [194]
81	Posts	Structure your project properly [75], [194]
82	Papers/posts	Use agile Practices [3], [9], [14], [49], [61], [64], [66], [76], [92], [103], [110], [115], [139]
83	Papers/posts	MLOps [14], [23], [47], [114], [119], [120], [121], [122], [195], [196], [197], [198], [199], [200]
84	Posts	DevOps [119], [120], [121], [196], [201]
85	Papers/posts	Team Data Science Process [14], [23], [76], [139], [202]
86	Papers/posts	Data-Driven Scrum [3], [139]
87	Papers/posts	CRISP-DM [3], [14], [58], [61], [114], [123], [139], [163], [202], [203]
88	Papers/posts	KDD [3], [58], [204], [205]

TABLE VIII
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE RELATED TO THE *DELIVERY AND RUNTIME* CATEGORY

ID	Source	Practices
89	Papers	Build a degraded sensitive model [113]
90	Papers/posts	Deploy and operationalize the model [13], [14], [18], [23], [47], [51], [56], [58], [65], [69], [75], [111], [114], [124], [125], [164], [165], [176]
91	Papers/posts	Continuous model monitoring [3], [18], [56], [58], [75], [79], [85], [86], [87], [90], [114], [125], [126], [168], [206]
92	Posts	Continuous testing [91], [114], [120]
93	Posts	Build specialized and reusable pipelines [13], [23], [65], [79], [86], [90], [114], [116], [118], [126]
94	Papers/posts	Continuous improvement and learning [37], [101], [114], [117], [119], [120], [196], [201]
95	Posts	Focus on infrastructure [80], [87], [118], [195]
96	Papers/posts	CI/CD [18], [122], [196], [197], [199]
97	Posts	Continuous success measures [83], [120], [121]
98	Posts	Serverless deployment [207]

conceptual map, plays a crucial role in guiding organizations to identify unfamiliar practices and bridge the gap between their current approaches and the expectations for the successful deployment of ML products. It provides a concrete perspective on the necessary investments and changes to transition towards a data-centric approach to AI development.

This article explores the ML product life cycle by analyzing existing concepts and practices documented in the literature. Considering our (complex) review protocol and specialized sources, we can shed light on the concentration of discussions surrounding ML products in specific knowledge areas. We found that data management and model management categories and technical practices associated with developing ML models are more consolidated, with only a few minor controversies in the literature. However, we have also identified the blurred frontiers between these areas when building the conceptual map, the interconnectedness, and interdependencies between different aspects of the field. For managers, our analysis has highlighted the importance of collaborating with data engineers and data scientists from the ML product strategy phase. Managers must have a solid knowledge of data analysis and how ML models make inferences to identify opportunities.

Our study has revealed the need for multidisciplinary research on adopting ML systematically. While we encountered numerous adoption case studies and valuable insights in the GL, which emphasized the interconnectedness of management decisions and the impact on ML development, we observed relatively limited research in academic literature. Other research opportunities identified are assessing the quality of ML products in organizations, qualifying managers for ML practice, addressing the social and ethical implications of ML, and adapting software engineering practices.

DATA AVAILABILITY

All the data related to this rese arch, supplying a chain of evidence, are available at https://github.com/alvesisaque/Practices-ML-Product. We also stored these data at https://doi.org/10.6084/m9.figshare.23502000.v1 for long-term archiving.

APPEDIX A
SEARCH QUERIES

Table III shows the search queries used in this study and presents the basic search terms and phrases used to effectively search for and retrieve relevant works for each category.

APPENDIX B
PRACTICES PRESENT IN THE ML PRODUCT LIFE CYCLE

Tables IV–VIII present the emerged practices, and we grouped them by the categories defined at the beginning of study design.

TABLE IX
DESCRIPTION OF PRACTICES RELATED TO PROBLEM DEFINITION AND SOLUTION DESIGN

ID	Method/Practice	Description
1	Business understanding and validation	Work with customers and stakeholders to identify business problems and formulate questions that data science techniques can address. This practice ensures the alignment of the product with the business goals. It is crucial throughout the project, particularly at the beginning of the product lifecycle, to ensure the solution meets business needs.
2	Identifying the right problem to solve	This involves determining real problems that people need help with by talking to them, looking through data, and observing behaviors. This process shifts the focus from technology-first to people-first. Causal stories are important for quantifying cause and effect relationships, performing root-cause analysis, and playing what-if scenarios. Domain knowledge is critical for achieving precision and accuracy in problem definition.
3	Verify how necessary is ML for the product	It involves assessing whether using ML is required to solve a problem, as it can be expensive and time-consuming. The project's initial phase should include studies to validate if ML is necessary, particularly if the business rules and data are too complex for human understanding and development without ML. This practice helps ensure that ML is only used when it is the best solution and justifies the investment of time and resources.
4	Define the role of ML on product	It involves determining the importance of ML in the project and its positive and negative impact. At the beginning of the project, it is essential to determine whether ML is a core or just a feature of the product and whether it adds unique value to solving a real problem. It is important to question whether adding ML to the product will improve it and whether to use ML to automate a task or augment a person's ability to do it.
5	Statement of expectation and intention	It involves being clear with the team about what the project can and cannot do and managing expectations carefully. During the idea design phase, clarifying how the product will influence customers' workflow and its unique selling proposition is important. All stakeholders should agree on the problem and the proposed solution.
6	Establish what is the outcome and what the data can offer	This practice involves identifying the project's desired outcome and determining the available data to achieve that outcome. Starting with a problem or available data is important to guide this process.
7	Set expectations for adaptation	Setting expectations for adaptation involves designing products that can adjust their responses based on user input over time. It requires understanding the context of the user's relationship with the AI product, how people currently solve problems, and how ML can ensure execution and adapt to changes in user behavior. Development can proceed with clear expectations and intentions by establishing what the product does and sharing it with the team.
8	Apply sign thinking to understand the user deeper	This practice involves applying an iterative process of Design Thinking to gain a deeper understanding of the user, challenge assumptions, and redefine problems to identify alternative strategies and solutions that may not be initially apparent with our initial understanding.
9	Lean Canvas	Lean Canvas is a business model template that helps to validate and visualize ideas by focusing on the key drivers of a business, such as customer segments, value proposition, revenue streams, and cost structure. It assists in operationalizing value creation from data by linking business objectives with technical implementation, ultimately helping to understand the potential business opportunity.
10	Apply prototyping to validate and align the objectives	The practice of prototyping involves creating experimental versions of a product or interface in order to validate and align project objectives. It is a useful tool for testing and refining ideas before full implementation, especially in the context of interfaces and user experiences.
11	Define the desired outcome	This practice involves defining the desired outcome of an ML project or product and ensuring that the results are aligned with the business objectives. It requires making key decisions about what the project tries to predict or identify and designing a reward function that encourages the right outcomes. This should be a collaborative process involving multiple perspectives and should involve careful consideration of the potential outcomes and potential pitfalls.
12	Translate user needs into data needs	Translating user needs into data needs means identifying the necessary data to reflect the users' needs and ensuring that the data used for training the AI system is representative and of high quality. This involves considering the scope of features, labeling quality, and the representativeness of the training dataset to ensure the AI system performs well and meets user needs.
13	AI companies need to create an ontology	Creating an ontology involves consistently representing data and data relationships within a business or organization. It helps to understand and organize data, especially when unstructured or scattered in different silos. It can be seen as an organization and documentation of data, similar to a data dictionary.
14	Conduct preliminary experiments and track	The process of evaluating multiple ML algorithms in parallel to determine the best model based on accuracy and organizing and tracking model training information across multiple runs with different configurations. This practice helps define a project's scope and feasibility and is useful in requirements validation. Experiment tracking is a part of MLOps that helps benchmark different models and configurations, making it easier to track progress and improve.
15	Include a goal description on requirements	Including a goal description in requirements means describing the expected outcome for the problem at hand, considering the specific data used. Doing so makes it possible to validate the requirements and select the best algorithm to solve the problem. This practice is closely related to conducting preliminary experiments and helps engineers to understand the problem better and develop software to select the best algorithm for the given data.
16	Reengineer some business processes	It involves analyzing and changing existing processes in order to optimize them for the use of ML or other advanced technologies. This practice may require a shift in traditional ways of doing things to fully take advantage of the potential benefits offered by these technologies.

APPENDIX C
PRACTICES AND DESCRIPTIONS

Tables IX–XIV provide a glossary of the methods and practices and their descriptions or definitions, offering a deeper understanding of their meaning.

APPENDIX D
CONCEPTUAL MAP

Fig. 16 presents, in high resolution, the complete conceptual map of this study. We encourage readers to zoom-in to explore the details of the complete conceptual map.

TABLE X
GLOSSARY OF METHODS AND PRACTICES RELATED TO PRODUCT MANAGEMENT

ID	Method/Practice	Description
17	Improvement Using User Feedback	It involves leveraging user feedback to enhance the user experience, improve technology, and provide personalized content. This can include receiving explicit feedback from prototype users and implicit feedback from end users to make continuous improvements to the product or service.
18	Working closely with consumers and stakeholders from day one	This practice involves working closely with consumers and stakeholders from the beginning of the project to ensure that their needs and expectations are met. It includes frequent communication with them to define the questions and scenarios, explaining findings in simple terms to non-experts, and having multiple iterations throughout the pipeline to improve the project based on feedback and testing.
19	Learn From Retrospective Meetings and Logs	This practice involves using retrospective meetings and logs to learn from past experiences and improve future work. By reflecting on past successes and failures, teams can identify areas for improvement and implement changes to achieve better outcomes in the future. These meetings can be used to plan new products and processes and identify best practices for future projects.
20	Review the Literature	It involves keeping up with the latest research, publications, and industry trends related to the specific problem or area of expertise. This practice helps in understanding the new technologies, models, and practices that best fit the problem and staying up-to-date with the latest advancements in the field. It also involves reading up on existing approaches, datasets, and other resources that may be relevant to the methodology used for building the model.
21	Map existing workflows	It is a practice of documenting the current processes involved in completing a task or achieving a goal. This helps identify improvement areas, inefficiencies, and opportunities for automation or augmentation. By mapping the workflow, stakeholders can better understand the necessary steps and make more informed decisions about how to optimize or redesign the process.
22	Documentation	It is storing and sharing information about a process or product. It involves documenting processes, experiment results, and other relevant information throughout the execution of a project. This helps ensure that knowledge is retained and easily accessible for future reference and use.
23	Balanced Scorecard	It is a strategic management tool that helps align objectives with project evolution and monitor risks by defining delivery dates and assessing status. It also assists in maintaining clear communication and aligning cross-functional teams on project milestones and tasks. Additionally, it helps evaluate customer expectations and assess the risk of negative impacts such as bad press, biased models, or violations of laws or company policies.
24	Calculate the cost and benefits	This practice involves assessing the financial and non-financial impacts of implementing a product or technology, such as AI and ML. This assessment helps determine if the benefits of building the product or using the technology outweigh the costs. It can also help identify potential risks and opportunities for improvement.
25	Feedback Loops	Feedback loops are a way to gather information on how a machine learning model performs and use that information to improve the model. This involves collecting data from users and stakeholders, analyzing that data, and using the insights gained to adjust the model. Feedback loops can also help measure the impact of a model and improve overall usability.
26	Manage the experiments	It is the practice of organizing and keeping track of the numerous experiments performed while developing machine learning models to identify the optimal model. This practice involves documenting experiment details, results, and metrics to understand the project's evolution better and make informed decisions. It is particularly important in the early stages of a project to ensure that experiments are properly managed and tracked.
27	Weighing the cost of false positives and false negatives	It involves calculating the risks and costs associated with having an incorrect outcome, which can be dangerous and expensive for the product or company. It is important to consider the consequences of false positives and false negatives when defining the reward function for your machine learning model and to weigh them differently based on their impact on users. Confidence indicators can also be included to mitigate the negative effects of these types of errors.
28	Evaluate Results, Define Metrics and Baselines	This practice involves establishing a baseline model that provides reasonable results with minimal effort, defining metrics to measure the success or failure of the product, and evaluating the results against the baseline. Establishing a baseline allows for a better understanding of the model's limitations, and identifying areas where a more complex model may be necessary. Defining metrics early helps track progress toward goals and measure the impact of the model.
29	Ensure to use it responsibly	This practice involves responsibly using machine learning by considering the impact of the technology and providing transparency to users about how their data is being used. It also involves adhering to data regulations and principles and considering the potential consequences of using machine learning in society.
30	Establish and deal with customer privacy information	This practice involves establishing and dealing with customer privacy information, which includes complying with government regulations, informing customers about the costs and benefits of sharing their data, asking for customer consent before harvesting their personal information, and penalizing non-compliant companies. It is important to protect customer privacy and data and to use their information only fairly and responsibly.
31	Detect and resolve fairness / Design with the fairness in mind / Commit to fairness	The practice of detecting and resolving fairness in ML involves designing with fairness in mind, using diverse data that reflect the context of the users, exploring datasets to understand biases, and monitoring the outcome to identify and address any fairness issues. Identifying possible biases and cleaning the model to avoid replicating them while optimizing the system for the values we care about is important.
32	Diagnose the value-destruction potential of a business	This practice involves assessing the potential negative impact of implementing machine learning on a business. It is important to diagnose the risk of ML characteristics on the solution and analyze how they may affect the business value. Regularly monitor impact metrics and identify potential negative outcomes to avoid any destruction of value. The practice also includes setting standards and guidelines for the team to address negative outcomes and improve user experience.
33	Control and observe the fairness, bias, and errors	Controlling and observing fairness, bias, and errors in ML involves identifying risks and creating contingency plans to prevent wrong decisions and environmental changes. It requires monitoring and ensuring that the application is accessible and not discriminatory while considering bias in the data collection and evaluation. The goal is to construct an ethical ML application by listing risks, identifying their severity, and weighing the cost of getting them wrong.
34	Explain and give to customers the control about how they are using and sharing their data	This practice involves informing users about how their data is used and giving them control over sharing it. By being transparent and clear about the benefits and risks of sharing their data, users are more likely to feel comfortable and willing to share information. Allowing customers to change their data-sharing preferences can also improve trust and forgiveness in case of data privacy breaches.
35	Manage privacy and security	Managing privacy and security involves implementing policies and procedures to protect personal data from unauthorized access, use, or disclosure. This includes being clear about data practices, reviewing data for sensitive information, consulting with legal experts, setting up infrastructure for privacy protection, and taking extra steps to anonymize data when necessary.

TABLE XI
DESCRIPTION OF PRACTICES RELATED TO THE DATA MANAGEMENT CATEGORY

ID	Method/Practice	Description
36	Create a dataset specification	Creating a dataset specification involves documenting the data needs, including format, properties, labels, and potential sources. This helps ensure that the data collected is relevant and sufficient for solving the defined problem statement based on user needs.
37	Document the data	Documenting the data involves recording the dataset's sources, transformations, history, and recommended uses. This documentation helps when sharing, reviewing, comparing, and using the data responsibly. Data Cards are a form of documentation that answer questions about what the data represents, where it comes from, how it was prepared, and how it should be used responsibly.
38	Data exploration: Know and study the data	Data exploration involves studying and understanding the data through statistical and visualization methods to identify patterns, correlations, and potential problems in the dataset. It helps decide which models or algorithms to use in subsequent steps and can reduce bias. Data visualization tools such as graphs and charts make data more understandable and enable easier interpretation of data.
39	Define the Data Strategy	Defining a data strategy is crucial for the success of an ML project. It involves planning and organizing data collection, preparation, and availability to avoid poor data choices and downstream effects. It is important to consider the advantages of using this data compared to competitors and ensure that the company understands the nature of ML products, their pros and cons, trade-offs, and potential consequences and costs of wrong predictions.
40	Define the Data Pipeline	Defining the data pipeline means establishing a process for collecting, cleaning, labeling, and evaluating data quality. It involves setting up a data quality control process with quality metrics, determining data labeling and cleaning methods, and evaluating the trade-offs between quality assurance cost and accuracy.
41	Prepare a data maintenance plan	It involves considering the long-term sustainability of a dataset and implementing strategies for preventing problems before they occur, preserving the dataset as the real world changes, and fixing errors that arise. This includes storing the dataset in a stable repository, deciding which properties should be preserved, keeping the data updated over time, and maintaining a detailed log of changes made to the dataset.
42	Ensure the Reliability, Availability, and Quality of Data	This practice involves focusing on obtaining more and better data and exploring it to identify and correct errors. This ensures that the data is suitable for the intended purpose and will succeed in real-world contexts. The goal is to identify the right data type and distribution to ensure the product's success.
43	Validate and test dataset to ensure integrity	This practice involves validating and testing the dataset to ensure its integrity, including checking for missing values, data standardization, identifying anomalous patterns, and mislabeled data. Different types of validation can be performed depending on the objectives and constraints of the project. The output from this process should be informative enough for data engineers to take action, and it needs high precision to avoid false alarms. In production, the model gets retrained periodically with new data, and the same data is used for retraining to ensure that the model adapts to changes in data characteristics. This practice is important to ensure that the model accuracy does not degrade over time due to erroneous data ingestion.
44	Data collection and evolution	Data collection and evolution is acquiring and updating datasets for machine learning models. It involves determining whether existing datasets can be reused, selecting suitable training data, determining the frequency and amount of data required, and ensuring the quality and comprehensiveness of the data.
45	Data cleaning and structuring	Data cleaning and structuring removes inaccurate, noisy, or incomplete data from a dataset and organizes the remaining data in a structured format suitable for analysis and modeling. It helps improve data quality and reduce bias, leading to better model performance and more accurate insights.
46	Data preprocessing	It is the process of cleaning, transforming, and preparing data for analysis or training machine learning models. It involves handling missing or erroneous data, scaling and normalizing features, encoding categorical variables, and other techniques to improve data quality and make it suitable for machine learning algorithms.
47	Data Labeling	Data labeling is the process of assigning relevant tags or categories to data, which is crucial for supervised learning. It can be done through automated processes or by human labelers, such as users, generalists, or subject matter experts. Creating guidelines is important to ensure consistency in labeling.
48	Data Integrations	It refers to combining data from multiple sources to create a unified view of the data. This is often necessary to maintain organization and ensure all relevant data is available for analysis. The process involves mapping data elements from different sources and transforming them to ensure compatibility before integrating them.
49	Data versioning, tracking, and provenance	This maintains a historical record of changes to datasets and machine learning models. It involves version control to track changes to data and models, as well as tracking of metadata to understand how the data was created and processed. This practice ensures reproducibility and enables easy rollbacks in case of errors or changes in requirements.
50	Data transformation	Data transformation is the process of converting and manipulating data into a format that is appropriate for data analysis and machine learning. This step is often performed during data preprocessing. It involves tasks such as data cleaning, normalization, encoding, feature engineering, and more, depending on the specific needs of the model being developed. The goal of data transformation is to ensure that the data is in a consistent and usable format and effectively captures the relevant features and patterns the model is designed to detect.
51	Data Reuse	Data Reuse refers to the practice of utilizing existing datasets, either from previous experiments or from online sources, to reduce the amount of effort required in data collection. However, caution must be taken to ensure that the data accurately reflects the users' reality and that any potential biases are identified and addressed.
52	Feature extraction and selection	It is the practice of identifying and selecting the most relevant and informative features from raw data to be used in machine learning models. It aims to build a useful model by finding the best features that capture the patterns and relationships in the data. Feature extraction can involve creating new variables or features, while feature selection focuses on selecting a subset of the existing features to improve model performance and reduce complexity.
53	Feature engineering	It uses domain knowledge and data mining techniques to extract, transform, and select relevant features from raw data to create new features for potential use in machine learning models. This process involves handling missing values, converting categorical data into numerical form, correcting non-Gaussian distributions, identifying outliers, and scaling features. It can be an expensive and time-consuming process that can be done manually or automated. The goal is to improve the performance of machine learning algorithms by helping them understand the data and identify patterns.
54	Automate feature engineering process	Automating feature engineering involves using software tools and algorithms to extract and select relevant features from raw data automatically. This process can save time and effort and improve the accuracy and performance of machine learning models. Automated feature engineering tools use data mining techniques to extract and create new features, handle missing values, transform variables, find outliers, and scale features. By automating this process, machine learning practitioners can focus on other aspects of the modeling process and iterate more quickly towards better models.

TABLE XII
DESCRIPTION OF PRACTICES RELATED TO THE MODEL MANAGEMENT CATEGORY

ID	Method/Practice	Description
55	Research ML Libraries and frameworks to be used	Researching ML libraries and frameworks entails identifying and evaluating the available tools and resources for implementing machine learning solutions. This includes assessing popular solutions and their licensing terms. By studying existing research, one can improve their approach and select the most suitable libraries for their project's needs. This practice enables one to make informed decisions about which tools to use, considering features, documentation, community support, and compatibility with data and models.
56	Establish the model requirements	Establishing the model requirements involves determining the most suitable types of models for a given problem, considering factors such as the need for real-time training. This practice is carried out during the testing and requirements-gathering phases of the solution design process. It helps ensure that the selected model meets the project's specific needs and requirements.
57	Test multiple hypotheses	Testing multiple hypotheses involves evaluating different potential explanations or solutions to a problem using statistical methods. This practice estimates the confidence level of rejecting a null hypothesis and choosing an alternative hypothesis. In the context of machine learning, testing multiple hypotheses is used to quickly validate ideas during the product definition and planning stages and to ensure that models are tested against various scenarios to avoid overfitting and validate against bias.
58	Model training	It is a process in which the selected model is trained and tuned using the chosen features and labeled data. The goal is to find the best possible set of parameters that will allow the model to accurately predict outcomes on new data. The process involves training the model on clean, collected data and labels, and tuning its parameters to optimize its performance. Monitoring the training runtime and considering whether realtime training is necessary for solving the problem is important.
59	Periodically retrain the model to prevent model staleness	Periodically retraining the model is a practice of regularly updating the machine learning model to prevent model staleness and ensure that it continues to perform accurately over time. It involves monitoring the model's performance on new data, identifying any degradation in performance, and retraining the model to improve its accuracy. This helps to prevent the model from becoming outdated and ensures that it continues to provide reliable predictions or classifications.
60	Balance underfitting and overfitting	This practice involves finding a balance between an ML model that is too simple and cannot capture complex relationships in the data and a model that is too complex and fits too closely to the training data, resulting in poor performance on new or test data. It is important to ensure that the training data is appropriate for the context and that the model is properly tuned to prevent overfitting or underfitting. This practice aims to optimize model performance and generalization to new data.
61	Interpretability: Understand how the model works	The practice of interpretability in ML involves ensuring that the model's decision-making process is transparent and understandable to humans. This can involve a trade-off between interpretability and accuracy, but ensuring that the model's decisions are not influenced by hidden biases or unfairness is important. By making the model explain how it arrived at a decision, practitioners can obtain insights and identify potential issues that might otherwise go unnoticed.
62	Avoid overfitting	This practice involves preventing an ML model from becoming too closely tailored to the training data and losing its ability to make accurate predictions on new data. Techniques such as testing and validating the dataset, dividing the dataset into training and testing sets, and comparing the accuracy of models based on their ability to predict observations not used in model development are commonly used to avoid overfitting. Overfitting can occur when a model memorizes the training set too closely and loses its ability to generalize and make predictions for new data.
63	Cross validation	It is a technique used to evaluate a machine learning model by dividing the dataset into several subsets and repeatedly training and testing the model on different combinations of these subsets. This helps to ensure that the model is not overfitting or underfitting the data and provides a more accurate measure of the model's performance.
64	Apply regularization	It is used to prevent overfitting by penalizing the model's complexity. It helps to balance a flexible and a conservative model by telling the model not to jump to conclusions and avoiding learning from too much noise. Regularization encompasses different techniques that penalize the model's weights, such as regularization, dropout, and early stopping.
65	Consider precision and recall tradeoffs	Considering precision and recall tradeoffs involves balancing the proportion of true positives correctly categorized with the number of false negatives and false positives to optimize the model's performance. The higher the precision, the fewer false positives, but the tradeoff is that there may be more false negatives. Conversely, the higher the recall, the fewer false negatives there are, but the tradeoff is that there may be more false positives. When optimizing these metrics, it is important to consider the tradeoffs between speed and accuracy.
66	Model validation	Ensures that the ML models are built in a way that the outcomes are predictable, consistent, and repeatable. It involves measuring various performance metrics, such as accuracy, precision, recall, and F1 score, to ensure that the model performs well on both the training and unseen data. It is important to validate the model during feature engineering and training to ensure it is not overfitting or underfitting the data and can make accurate predictions on new data.
67	Model evaluation and result interpretation	Refer to the process of measuring the performance of a trained ML model on test or validation datasets using pre-defined evaluation metrics and interpreting the results to understand the model's behavior and make informed decisions based on the outcomes. It involves assessing the significance and practicality of the model's results and considering domain-specific factors to determine the usefulness and relevance of the model's predictions.
68	Assessing prediction accuracy	It measures the quality of the predictions made by a model using various metrics and techniques. It involves evaluating the model's performance on test data using pre-defined evaluation measures, such as mean square error, means absolute deviation, and mean absolute percentage error for regression models, and metrics like accuracy, precision, recall, and specificity for classification models. The aim is to ensure that the model is accurate, consistent, and reliable and can be trusted to make accurate predictions.
69	Control and evaluate model peformance	This practice involves monitoring and controlling the execution time and processing information of the model. It may take a long time to train the model depending on the chosen model and the number of experiments to be performed. Therefore, it is important to simplify the model or reduce the data while maintaining the quality defined in the metrics. It may also be necessary to consider hardware upgrades to improve the model's performance. Additionally, the practice includes evaluating the model's performance and making any necessary adjustments to improve its accuracy and efficiency.
70	Tune the models and hyperparameters	It involves adjusting the settings and configurations of an AI model to achieve the best possible performance for a specific use case. This includes tuning the model's fixed parameters and hyperparameters, such as learning rates and regularization coefficients, to optimize the model's accuracy and generalization ability.

TABLE XIII
GLOSSARY OF METHODS AND PRACTICES RELATED TO THE SOFTWARE ENGINEERING CATEGORY

ID	Method/Practice	Description
71	Test Early and frequently from end to end	Refers to validating and testing each component and stage of the ML pipeline, from data collection and preprocessing to model training and deployment, to ensure that everything is working as intended and to identify and fix any issues or errors as soon as possible. This helps improve the quality and reliability of the machine learning system and reduces the time and costs associated with debugging and fixing problems later in the development process.
72	Ensure code and architecture reusability	Involves designing code and architecture to allow easy reuse and adaptation in future projects or tasks. By creating modular and well-organized code, functions, and algorithms, practitioners can save time and effort in future projects by reusing existing solutions. This can include sharing classes and functions for common tasks such as data cleaning, feature extraction, hyperparameter tuning, building statistical models, and reusing pre-trained models.
73	Modularizing train code	Refers to breaking down the code of an ML model into separate modules that perform specific functions. This makes the code easier to understand, test, and maintain, allowing better collaboration between team members. The concept of modularity is important in both traditional software development and machine learning. However, machine learning can also involve dependencies between modules based on large amounts of data.
74	Version control	Version control in ML involves keeping track of the changes made to the model configurations and experiment metadata during the experimentation process. This helps maintain a record of the experiments conducted with various configurations and environments and allows for easy retrieval of models with different configurations. Using version control ensures that the ML project remains organized and efficient, reducing the risk of losing important data.
75	Use agile Practices	Using agile practices in data science projects involves continuous collaboration and communication between data scientists and stakeholders, frequent reviews and direction from the business owner, and the production of evolving models and releases that are very user-focused. Agile practices help ensure that high-level requirements are agreed upon early in the development lifecycle and are reviewed and updated as the project progresses, leading to better outcomes and a stronger sense of ownership for the business owner.
76	MLOps	MLOps promotes collaboration and communication between data scientists and operations professionals to enhance ML model quality, simplify management, and automate deployment in large-scale production environments. It brings agility, aligns models with business needs and regulations, and adopts DevOps practices. MLOps addresses challenges in software production, requires a hybrid team, and considers performance degradation and Training-Serving Skew.
77	DevOps	DevOps is a collaborative and culture-focused approach in software engineering that promotes effective communication and collaboration between software developers and operations teams. It emphasizes automation, continuous integration, and continuous delivery to facilitate faster and more reliable software releases while aligning closely with business goals.
78	CRISP-DM	CRISP-DM is a data science methodology that describes the entire data science process in six phases. It provides a blueprint for conducting data science projects, from planning to deployment. The methodology encourages interoperable tooling and iterating quickly to provide more value. However, it lacks communication strategies with stakeholders and should be combined with a project management approach such as Agile methodologies like Kanban or Scrum.
79	TDSP	TDSP is an agile and iterative data science method developed by Microsoft, combining elements of Scrum and CRISP-DM. It includes five iterative stages: Business Understanding, Data Acquisition and Understanding, Modeling, Deployment, and Customer Acceptance. TDSP aims to deliver predictive analytics solutions and intelligent applications efficiently, improve team collaboration and learning, and provide guidance on the tasks needed to use predictive models.
80	DDS	DDS is an agile method designed specifically for data science projects, allowing for lean and iterative exploratory data analysis. It allows for capability-based iterations and flexible meeting schedules but requires high-level item estimation. DDS provides a framework for prioritizing potential tasks and improving team performance.
81	KDD	Knowledge Discovery in Databases (KDD) is a process of discovering useful patterns and insights from large datasets. It involves multiple steps, including developing an understanding of the application domain, cleaning and preprocessing data, choosing appropriate data mining tasks and algorithms, mining data, interpreting patterns, and consolidating knowledge. The ultimate goal of KDD is to identify valid, novel, potentially useful, and understandable patterns in data.
82	Ensemble learning	It is a machine learning technique that involves combining multiple models to improve prediction accuracy and generalization performance beyond what can be achieved by a single model.
83	Create a controled environment	Creating a controlled environment involves setting up a reproducible and isolated environment for ML projects. This can be achieved using tools like Docker to containerize the environment, version control to manage changes, and keeping the source code and dataset separate. A controlled environment helps with evaluation, especially in ensemble models or modularized code cases. It also helps maintain consistency and reproducibility throughout the development process.
84	Increase the computational capability and take advantage of the latest hardware	This practice involves increasing the computational capability to handle large amounts of data taking advantage of the latest hardware technology to optimize the performance of ML models. It is important to improve and make learning faster and more practical constantly. Many companies are adopting this practice and investing in the capability to train and test models multiple times a day.
85	Automation	Automation in machine learning streamlines tasks like data preprocessing, feature engineering, model training, and deployment. It boosts efficiency, reduces errors, and enables faster iterations. Automated deployment automates retraining and deployment, ensuring scalability and optimal performance.
86	Keep the infrastructure right	Keeping the infrastructure right involves ensuring a well-tested and validated system infrastructure. Focusing on infrastructure for the first pipeline and testing it independently from machine learning is important. This helps avoid unexpected infrastructure issues and ensures a stable environment for machine learning.
87	Structure your project properly	It involves creating a well-organized directory and file structure, saving the versions of parameters and models, documents, licenses, Jupyter notebooks, and other relevant information. This improves teamwork and makes it easier for different team members to focus on their tasks. A clean and organized structure helps avoid future problems and ensures the smooth running of the project.

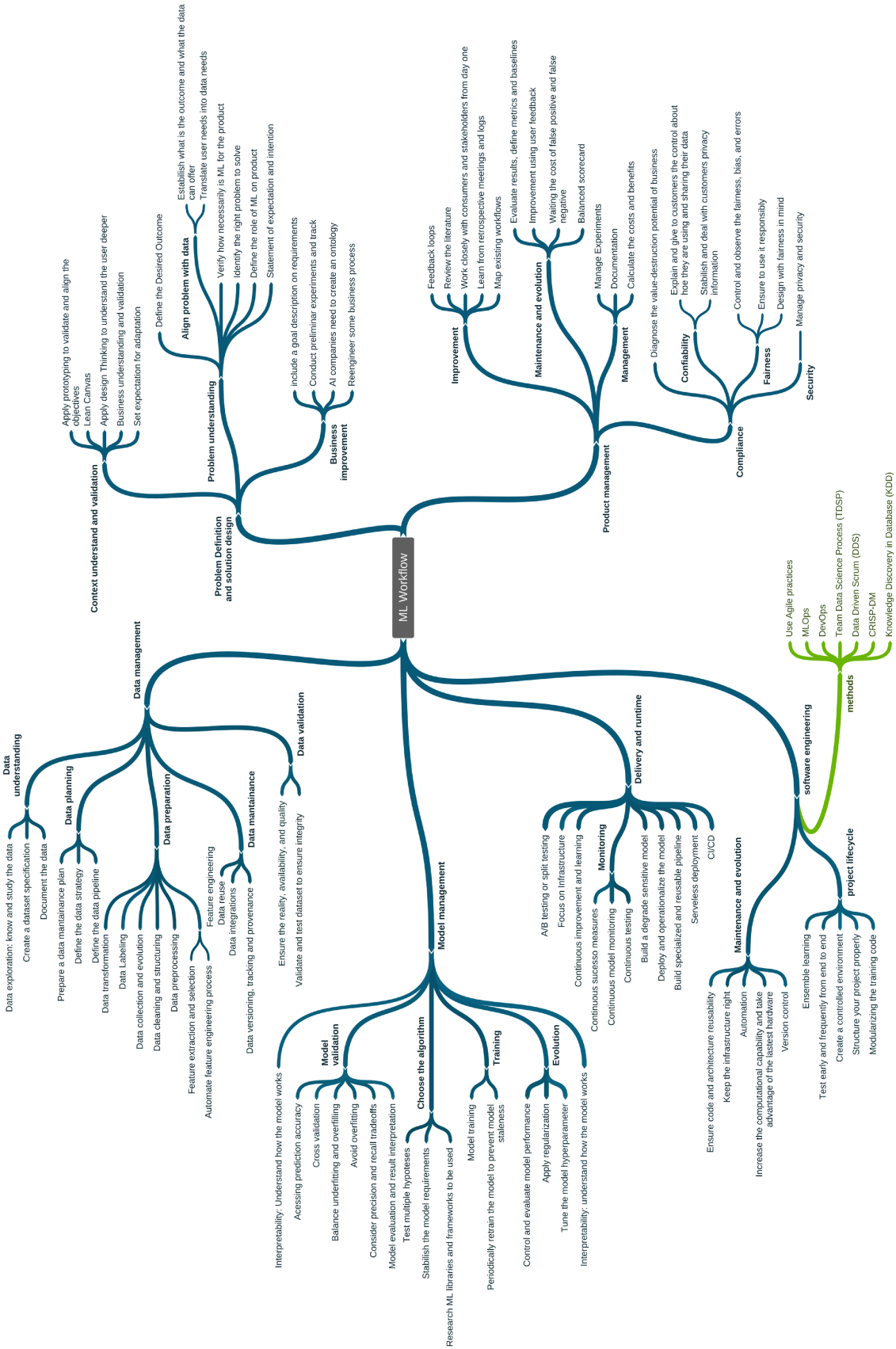

Fig. 16. Conceptual map displaying the various practices and their interrelationships related to effectively managing ML systems.

TABLE XIV
GLOSSARY OF METHODS AND PRACTICES RELATED TO THE DELIVERY AND RUNTIME CATEGORY

ID	Method/Practice	Description
88	Build a degraded sensitive model	This practice involves building a ML model capable of identifying a reduction in the quality of its results and adapting to maintain or improve its performance, even when faced with new or changing data. It is important to address potential performance degradation issues that may occur over time in machine learning products.
89	Deploy and operationalize the model	Deploying and operationalizing the model involves integrating predictive models into software products and systems, continuously feeding new data, retraining models, and checking for performance. It requires a versioning system to handle model parameters, configuration, datasets, and feature pipelines. Models are exposed with an open API interface for real-time or batch predictions, depending on the business requirements.
90	Continuous model monitoring	Continuous model monitoring is the practice of monitoring the performance of ML models in production, identifying potential errors or issues, and updating the model or taking corrective actions as needed. It involves tracking the model's performance over time and monitoring the data used to build it to ensure it remains relevant and accurate. Continuous monitoring is essential to maintain the scalability and robustness of ML models in production.
91	Continuous testing	Continuous testing in the context of ML involves constantly testing and validating the ML models in production to identify possible errors and ensure that the models are performing as expected. This is particularly important because real-world data is prone to shifts and distribution changes affecting the model's performance. Continuous testing helps ensure the models are still valid and accurate and provides opportunities to retrain or update the models if needed.
92	Build specialized and reusable pipelines	This practice involves building specialized pipelines for the company's specific context to facilitate the loading and editing of data and experimentation with different algorithm permutations. It also emphasizes decoupling the model from infrastructure components for easier updates. The goal is to create reusable pipelines that can be used consistently and efficiently.
93	Continuous improvement and learning	Continuous improvement and learning is a method in which an ML system is trained frequently and autonomously to improve its performance over time. The system learns from new data and feedback, adapting and evolving as needed. This method involves regularly updating the model, retraining it with new data, and implementing necessary changes to improve performance. The ultimate goal is to create a self-learning and self-improving system that can adapt to new challenges and maintain high performance over time.
94	Focus on infrastructure	This practice involves considering infrastructure as a critical aspect from the beginning of the project. It means ensuring that the appropriate hardware, software, and network components are in place to support ML models' development, training, and deployment. A reliable infrastructure ensures the ML system runs smoothly, efficiently, and accurately, reducing downtime and increasing productivity.
95	CI/CD	CI/CD stands for Continuous Integration/Continuous Delivery, and it is a set of practices and techniques used to ensure that software products are developed and delivered quickly and reliably. It involves automating the build, test, and deployment processes and constantly integrating new code into the main codebase. This helps to catch errors and conflicts early on, which in turn helps to improve the quality of the software product and reduce the time to market.
96	Continuous success measures	It refers to continuously measuring an ML system's success during its delivery and run time to ensure that it is meeting the desired outcomes and goals. This involves setting up metrics and monitoring processes to track the system's performance in real-world scenarios and using this feedback to make ongoing improvements and optimizations to the system. Ultimately, the goal is to achieve and maintain high performance, reliability, and user satisfaction levels throughout the ML system's full delivery and run time.
97	Serverless deployment	Serverless deployment is a cloud computing model where servers are hidden from view and managed by a provider. This allows developers to focus on building and deploying their applications without worrying about server configuration. In machine learning, serverless deployment can save time and effort in retraining models as it allows for constant adjustments based on real-world data.
98	A/B testing or Split Testing	A/B testing or Split Testing compares two or more versions of a product or workflow to determine which one performs better. It involves randomly assigning users or data points to different versions and measuring their performance against predefined key performance indicator (KPIs).This technique is commonly used in software development and online services to improve user experience and product effectiveness.

REFERENCES

[1] S. Amershi et al., "Software engineering for machine learning: A case study," in *Proc. IEEE/ACM 41st Int. Conf. Softw. Eng., Softw. Eng. Pract.*, 2019, pp. 291–300.

[2] Z. Wan, X. Xia, D. Lo, and G. C. Murphy, "How does machine learning change software development practices?," *IEEE Trans. Softw. Eng.*, vol. 47, no. 9, pp. 1857–1871, Sep. 2021.

[3] M. Hesenius, N. Schwenzfeier, O. Meyer, W. Koop, and V. Gruhn, "Towards a software engineering process for developing data-driven applications," in *Proc. IEEE/ACM 7th Int. Workshop Realizing Artif. Intell. Synergies Softw. Eng.*, 2019, pp. 35–41.

[4] R. Akkiraju et al., "Characterizing machine learning processes: A maturity framework," in *Proc. Int. Conf. Bus. Process Manage.*, 2020, pp. 17–31.

[5] R. V. Regalado, "Building artificial intelligent (AI) products that make sense," in *Proc. 4th Int. Conf. Human-Comput. Interact. User Experience Indonesia*, 2018, pp. 93–96.

[6] R. Shams, "Developing machine learning products better and faster at startups," *IEEE Eng. Manage. Rev.*, vol. 46, no. 3, pp. 36–39, Third Quarter, 2018.

[7] J. Hodson, "How to make your company machine learning ready," *Harvard Bus. Rev.*, 2016. [Online]. Available: https://hbr.org/2016/11/how-to-make-your-company-machine-learning-ready

[8] I. Lee and Y. J. Shin, "Machine learning for enterprises: Applications, algorithm selection, and challenges," *Bus. Horiz.*, vol. 63, no. 2, pp. 157–170, 2020.

[9] K. Singla, J. Bose, and C. Naik, "Analysis of software engineering for agile machine learning projects," in *Proc. IEEE 15th India Council Int. Conf.*, 2018, pp. 1–5.

[10] S. Wang et al., "Machine/deep learning for software engineering: A systematic literature review," *IEEE Trans. Softw. Eng.*, vol. 49, no. 3, pp. 1188–1231, Mar. 2023.

[11] S. Schelter, F. Biessmann, T. Januschowski, D. Salinas, S. Seufert, and G. Szarvas, "On challenges in machine learning model management," *IEEE Data Eng. Bull.*, 2015. [Online]. Available: https://www.amazon.science/publications/on-challenges-in-machine-learning-model-management

[12] Y. Roh, G. Heo, and S. E. Whang, "A survey on data collection for machine learning: A big data—AI integration perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1328–1347, Apr. 2021.

[13] G. Gharibi, V. Walunj, R. Nekadi, R. Marri, and Y. Lee, "Automated end-to-end management of the modeling lifecycle in deep learning," *Empirical Softw. Eng.*, vol. 26, no. 2, pp. 1–33, 2021.

[14] I. Martinez, E. Viles, and I. G. Olaizola, "Data science methodologies: Current challenges and future approaches," *Big Data Res.*, vol. 24, 2021, Art. no. 100183. 3

[15] Google, "MLOps: Continuous delivery and automation pipelines in machine learning," White paper Google Cloud, 2020. Accessed: Apr.2020. [Online]. Available: https://cloud.google.com/solutions/machine-learning/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

[16] D. Sculley et al., "Hidden technical debt in machine learning systems," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 2503–2511.

[17] M. Wen, L. Leite, F. Kon, and P. Meirelles, "Understanding FLOSS through community publications: Strategies for grey literature review," in *Proc. IEEE/ACM 42nd Int. Conf. Softw. Eng., New Ideas Emerg. Results*, 2020, pp. 89–92.

[18] A. Serban, K. Van Der Blom, H. Hoos, and J. Visser, "Adoption and effects of software engineering best practices in machine learning," in *Proc. IEEE/ACM 14th Int. Symp. Empirical Softw. Eng. Meas.*, 2020, pp. 1–12.

[19] A. C. Serban and J. Visser, "Adapting software architectures to machine learning challenges," in *Proc. IEEE Int. Conf. Softw. Anal., Evol. Reeng.*, 2022, pp. 152–163.

[20] D. Muiruri, L. E. Lwakatare, J. K. Nurminen, and T. Mikkonen, "Practices and infrastructures for machine learning systems: An interview study in finnish organizations," *Computer*, vol. 55, no. 6, pp. 18–29, 2022.

[21] M. Dilhara, A. Ketkar, and D. Dig, "Understanding software-2.0: A study of machine learning library usage and evolution," in *ACM Trans. Softw. Eng. Methodol.*, vol. 30, no. 4, pp. 1–42, 2020.

[22] T. H. Davenport and R. Ronanki, "Artificial intelligence for the real world," *Harvard Bus. Rev.*, vol. 96, no. 1, pp. 108–116, 2018.

[23] "The team data science process lifecycle," Blog post at Microsoft, 2020. Accessed: Jun. 2021. [Online]. Available: https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/lifecycle

[24] D. A. Tamburri, F. Palomba, and R. Kazman, "Success and failure in software engineering: A followup systematic literature review," *IEEE Trans. Eng. Manage.*, vol. 68, no. 2, pp. 599–611, Apr. 2021.

[25] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA, USA: MIT, 2020.

[26] P. Kriens and T. Verbelen, "What machine learning can learn from software modularity," *Computer*, vol. 55, no. 9, pp. 35–42, 2022.

[27] R. Wirth, "CRISP-DM: Towards a standard process model for data mining," in *Proc. 4th Int. Conf. Practical Appl. Knowl. Discov. Data Mining*, 2000, pp. 29–39.

[28] B. Nushi, E. Kamar, E. Horvitz, and D. Kossmann, "On human intellect and machine failures: Troubleshooting integrative machine learning systems," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1017–1025.

[29] J. Anon and C. G. de Villaumbrosia, *The Product Book*. San Francisco, CA, USA: Product School, 2017.

[30] S. D. Eppinger and G. Geracie, *The Guide to the Product Management and Marketing Body of Knowledge*. Reno, NV, USA: Product Manage. Educ. Inst., 2013.

[31] O. Springer and J. Miler, "The role of a software product manager in various business environments," in *Proc. Federated Conf. Comput. Sci. Inf. Syst.*, 2018, pp. 985–994.

[32] R. Salay, R. Queiroz, and K. Czarnecki, "An analysis of ISO 26262: Using machine learning safely in automotive software," 2017, *arXiv:1709.02435*.

[33] V. Sridhar, S. Subramanian, D. Arteaga, S. Sundararaman, D. Roselli, and N. Talagala, "Model governance: Reducing the anarchy of production ML," in *Proc. Annu. Tech. Conf.*, 2018, pp. 351–358.

[34] G. Coleman and R. O'Connor, "Investigating software process in practice: A grounded theory perspective," *J. Syst. Softw.*, vol. 81, no. 5, pp. 772–784, 2008.

[35] S. J. Warnett and U. Zdun, "Architectural design decisions for the machine learning workflow," *Computer*, vol. 55, no. 3, pp. 40–51, 2022.

[36] T. Mikkonen, J. K. Nurminen, M. Raatikainen, I. Fronza, N. Mäkitalo, and T. Männistö, "Is machine learning software just software: A maintainability view," in *Proc. Int. Conf. Softw. Qual.*, 2021, pp. 94–105.

[37] A. Arpteg, B. Brinne, L. Crnkovic-Friis, and J. Bosch, "Software engineering challenges of deep learning," in *Proc. 44th Euromicro Conf. Softw. Eng. Adv. Appl.*, 2018, pp. 50–59.

[38] E. De Cristofaro, "A critical overview of privacy in machine learning," *IEEE Secur. Privacy*, vol. 19, no. 4, pp. 19–27, Jul./Aug. 2021.

[39] V. Garousi, M. Felderer, and M. V. Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," *Inf. Softw. Technol.*, vol. 106, pp. 101–121, 2019.

[40] H. R. Rothstein, A. J. Sutton, and M. Borenstein, *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*. Hoboken, NJ, USA: Wiley, 2005.

[41] A. Paez, "Grey literature: An important resource in systematic reviews," *J. Evidence-Based Med.*, vol. 10, no. 3, pp. 233–240, 2017 .

[42] P. Ralph et al., "Empirical standards for software engineering research," 2021, *arXiv:2010.03525v2*.

[43] K. Charmaz, "Grounded theory as an emergent method," *Handbook Emergent Methods*, vol. 155, pp. 155–172, 2008.

[44] H. Zhang, X. Zhou, X. Huang, H. Huang, and M. A. Babar, "An evidence-based inquiry into the use of grey literature in software engineering," *Proc. IEEE/ACM 42nd Int. Conf. Softw. Eng.*, 2020, pp. 1422–1434.

[45] R. J. Adams, P. Smart, and A. S. Huff, "Shades of grey: Guidelines for working with the grey literature in systematic reviews for management and organizational studies," *Int. J. Manage. Rev.*, vol. 19, no. 4, pp. 432–454, 2017.

[46] X. Zhou, "How to treat the use of grey literature in software engineering," in *Proc. Int. Conf. Softw. Syst. Process.*, 2020, pp. 189–192.

[47] K. Fanous, "Software 2.0 vs software 1.0," Blog post at Medium, 2020. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@karimfanous/software-2-0-vs-software-1-0-5df3f0d6d62a

[48] D. Budgen and P. Brereton, "Performing systematic literature reviews in software engineering," in *Proc. 28th Int. Conf. Softw. Eng.*, 2006, pp. 1051–1052.

[49] J. Zamora and P. Herrera, "How prepared is your business to make the most of IA," *Harvard Bus. Rev.*, 2018. [Online]. Available: https://store.hbr.org/product/how-prepared-is-your-business-to-make-the-most-of-ai/iir213?sku=IIR213-PDF-ENG

[50] B. Huang, "How to manage machine learning products," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/how-to-manage-machine-learning-products-part-1-386e7011258a

[51] L. Ampil, "Basics of data science product management: The ML workflow," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/basics-of-ai-product-management-orchestrating-the-ml-workflow-3601e0ff6baa

[52] C. Mewald, "Data: A key requirement for your machine learning (ML) product," Blog post at Medium, 2018. Accessed: Nov. 2020. [Online]. Available: https://medium.com/thelaunchpad/data-a-key-requirement-for-your-machine-learning-ml-product-9195ace977d4

[53] G. Chandrasekhar, "7 elements of ai product strategy," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/defining-your-ai-product-strategy-7-areas-of-focus-2cf112c82c07

[54] A. I. Canhoto and F. Clear, "Artificial intelligence and machine learning as business tools: A framework for diagnosing value destruction potential," *Bus. Horiz.*, vol. 63, no. 2, pp. 183–193, 2020.

[55] M. Rahimi, J. L. Guo, S. Kokaly, and M. Chechik, "Toward requirements specification for machine-learned components," in *Proc. IEEE 27th Int. Requirements Eng. Conf. Workshops*, 2019, pp. 241–244.

[56] E. D. S. Nascimento, I. Ahmed, E. Oliveira, M. P. Palheta, I. Steinmacher, and T. Conte, "Understanding development process of machine learning systems: Challenges and solutions," in *Proc. IEEE/ACM Int. Symp. Empirical Softw. Eng. Meas.*, 2019, pp. —6.

[57] S. Nalchigar, E. Yu, and K. Keshavjee, "Modeling machine learning requirements from three perspectives: A case report from the healthcare domain," *Requirements Eng.*, vol. 26, no. 2, pp. 237–254, 2021.

[58] O. Marbán, J. Segovia, E. Menasalvas, and C. Fernández-Baizán, "Toward data mining engineering: A software engineering approach," *Inf. Syst.*, vol. 34, no. 1, pp. 87–107, 2009.

[59] H. Ferreira, P. Ruivo, and C. Reis, "How do data scientists and managers influence machine learning value creation," *Procedia Comput. Sci.*, vol. 181, pp. 757–764, 2021.

[60] H. Takeuchi and S. Yamamoto, "Business analysis method for constructing business—AI alignment model," *Procedia Comput. Sci.*, vol. 176, pp. 1312–1321, 2020.

[61] K. Biesialska, X. Franch, and V. Muntés-Mulero, "Big data analytics in agile software development: A systematic mapping study," in *Inf. Softw. Technol.*, vol. 132, 2020, Art. no. 106448.

[62] V. Patha, "A product manager's guide to machine learning: Balanced scorecard," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/a-product-managers-guide-to-machine-learning-balanced-scorecard-344cd37ab4a7

[63] A. Mukherjee, "Product definition in the age of AI," Blog post at Towards Data Science, 2018. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/product-definition-in-the-age-of-ai-619a417e3415

[64] A. Pinhasi, "Machine learning products are built by teams, not unicorns," Blog post at Medium, 2019. Accessed: Dec. 2020. [Online]. Available: https://medium.com/@assaf.pinhasi/machine-learning-products-are-built-by-teams-not-unicorns-f3420c0e770b

[65] T. H. Davenport, E. Brynjolfsson, A. McAfee, and H. J. Wilson, *Artificial Intelligence: The Insights You Need From Harvard Business Review*. Brighton, MA, USA: Harvard Bus. Press, 2019.

[66] R. Sheng, "Managing data science as products," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020 [Online]. Available: https://towardsdatascience.com/managing-data-science-as-products-671077e625b3

[67] J. Rand, "Assessing the feasibility of a machine learning model," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/assessing-the-feasibility-of-a-machine-learning-model-ae36f4180f8

[68] D. G. ThomasH. Davenport, and A. Guda, "How to design an AI marketing strategy," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/07/how-to-design-an-ai-marketing-strategy

[69] M. Kim, T. Zimmermann, R. DeLine, and A. Begel, "The emerging role of data scientists on software development teams," in *Proc. IEEE/ACM 38th Int. Conf. Softw. Eng.*, 2016, pp. 96–107.

[70] A. R. Team, "User needs + defining success," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/user-needs/

[71] A. R. team, "Data collection + evaluation," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/data-collection/

[72] S. Pandian, "Understand machine learning and its end-to-end process," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/12/understand-machine-learning-and-its-end-to-end-process/

[73] D. Griffin, "Answering the big three data science questions at Cisco," Blog post at Cisco, 2020. Accessed: Jun. 2021. [Online]. Available: https://blogs.cisco.com/analytics-automation/answering-the-big-three-data-science-questions-at-cisco

[74] H. Yin, F. Fan, J. Zhang, H. Li, and T. F. Lau, "The importance of domain knowledge," Blog post at ML.CMU, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/1-domain-knowledge/

[75] N. Barla, "How to organize deep learning projects—Examples of best practices," Blog post at Neptune, 2021. Accessed: Jun. 2021. [Online]. Available: https://neptune.ai/blog/how-to-organize-deep-learning-projects-best-practices

[76] "Agile development of data science projects," Blog post at Microsoft, 2020. Accessed: Jun. 2021. [Online]. Available: https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/agile-development

[77] T. Stackpole, "How to set your AI project up for success," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/12/how-to-set-your-ai-project-up-for-success

[78] B. Huang, "Three questions every ML product manager must answer," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/three-questions-every-ml-product-manager-must-answer-35d73127cd5d

[79] "My framework for helping startups build and deploy data science," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/my-framework-for-helping-startups-build-and-deploy-data-science-43cf40bc1a1e

[80] E. Dezhic, "Machine learning rules in a nutshell," Blog post at Towards Data Science, 2018. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/machine-learning-rules-in-a-nutshell-dfc3e6839163

[81] U. Barlaskar, "Machine learning for product managers," Blog post at Towards Data Science, 2018. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/machine-learning-for-product-managers-part-i-problem-mapping-5436132c3a6e

[82] A. Mukherjee, "Lean startup and machine learning," Blog post at Towards Data Science, 2018. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/lean-startup-and-machine-learning-39a96efecb2a

[83] Y. Gavish, "The step-by-step pm guide to building machine learning based products," Blog post at Medium, 2017. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@yaelg/product-manager-pm-step-by-step-tutorial-building-machine-learning-products-ffa7817aa8ab

[84] J. Hurlock, "Building machine learning based products," Blog post at Medium, 2016. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@jonhurlock/building-machine-learning-based-products-part-i-an-introduction-to-ml-products-406235e53be3

[85] P. Pathak, "A product management framework for machine learning," Blog post at Medium, 2018. Accessed: Nov. 2020. [Online]. Available: https://productcoalition.com/my-framework-for-machine-learning-products-part-1-of-3-2354a0da63d2

[86] S. Fabri, "Creating ML and AI products—The smart products pipeline," Blog post at Medium, 2020. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@simon.fabri_25096/the-smart-products-pipeline-404472842bfb

[87] M. Zinkevich, "Rules of machine learning: Best practices for ML engineering," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://developers.google.com/machine-learning/guides/rules-of-ml

[88] J. Reilly, "How no-code platforms can bring AI to small and midsize businesses," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/11/how-no-code-platforms-can-bring-ai-to-small-and-midsize-businesses

[89] K. Holstein, J. W. Vaughan, H. Daumé III, M. Dudik, and H. Wallach, "Improving fairness in machine learning systems: What do industry practitioners need?," in *Proc. Conf. Hum. Factors Comput. Syst.*, 2019, pp. 1–16.

[90] R. Agarwal, "Six important steps to build a machine learning system," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/6-important-steps-to-build-a-machine-learning-system-d75e3b83686

[91] K. Schöfegger, "Lessons from building ML based products," Blog post at Medium, 2019. Accessed: Dec. 2020. [Online]. Available: https://medium.com/@kschoefegger/lessons-from-building-ml-based-products-4c7579724c91

[92] C. B. Canals, "How IA will—and won't—change the way you manage," *Harvard Bus. Rev.*, 2016. [Online]. Available: https://store.hbr.org/product/how-ai-will-and-won-t-change-the-way-you-manage-artificial-intelligence/iir211?sku=IIR211-PDF-ENG

[93] A. L. Åkerman, "Ask a techspert: How do machine learning models explain themselves?" Blog post at Google.ai, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.google/inside-google/googlers/ask-techspert-how-do-machine-learning-models-explain-themselves/

[94] People + AI Guidebook, "Mental models," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/mental-models/

[95] J. McKendrick, "AI adoption skyrocketed over the last 18 months," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/09/ai-adoption-skyrocketed-over-the-last-18-months

[96] Synthesized, "Shipping disruptive ML/AI products at scale (not papers). The engineer's take on scaling ML development processes," Blog post at Medium, 2020. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@synthesized/shipping-disruptive-ml-ai-products-at-scale-not-papers-4008c6131f12

[97] S. Stanford, "How to build accountability into your AI," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/08/how-to-build-accountability-into-your-ai

[98] B. Babic, I. G. Cohen, T. Evgeniou, and S. Gerke, "When machine learning goes off the rails a guide to managing the risks," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/01/when-machine-learning-goes-off-the-rails

[99] A. Fedyk, "How to tell if machine learning can solve your business problem," *Harvard Bus. Rev.*, vol. 5, 2016. [Online]. Available: https://hbr.org/2016/11/how-to-tell-if-machine-learning-can-solve-your-business-problem

[100] J. Rodrigues, "What's the role of designers while designing AI and ML products?" Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://medium.com/@rodriguesjohnbaptist/whats-the-role-of-designers-while-designing-ai-and-ml-products-70e344285175

[101] B. Gottipati, "Bringing design thinking to AI products," Blog post at Medium, 2019. Accessed: Dec. 2020. [Online]. Available: https://blog.usejournal.com/bringing-design-thinking-to-ai-products-f17386cf2ff2

[102] I. Metelskaia, O. Ignatyeva, S. Denef, and T. Samsonowa, "A business model template for AI solutions," in *Proc. Int. Conf. Intell. Sci. Technol.*, 2018, pp. 35–41.

[103] D. Lianoudakis, "The AI product owner's tool kit," Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://medium.com/@dlianou/the-ai-product-owners-tool-kit-aeaa2356f716

[104] L. E. Lwakatare, A. Raj, J. Bosch, H. H. Olsson, and I. Crnkovic, "A taxonomy of software engineering challenges for machine learning systems: An empirical investigation," in *Proc. Int. Conf. Agile Softw. Develop.*, 2019, pp. 227–243.

[105] O. Schoppe, "The role of design in machine learning," Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://medium.com/salesforce-ux/the-role-of-design-in-machine-learning-ae968ea90aac

[106] N. Polyzotis, S. Roy, S. E. Whang, and M. Zinkevich, "Data management challenges in production machine learning," in *Proc. ACM Int. Conf. Manage. Data*, 2017, pp. 1723–1726.

[107] V. Chaoji, R. Rastogi, and G. Roy, "Machine learning in the real world," *Proc. VLDB Endowment*, vol. 9, no. 13, pp. 1597–1600, 2016.

[108] P. Riley, "Good data analysis," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://developers.google.com/machine-learning/guides/good-data-analysis

[109] *Customer Data and Privacy*, Brighton, MA, USA: Harvard Bus. Press,, 2019.

[110] S. Gupta, "The key concept of scrum in machine learning," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/11/the-key-concept-of-scrum-in-machine-learning/

[111] W. Yandong and H. Kezheng, "Development of life originated artificial designer system," in *Proc. 3rd Int. Conf. Inf. Technol. Appl.*, 2005, vol. 1, pp. 319–324.

[112] S. Earley and J. Bernoff, "Is your data infrastructure ready for AI?," *Harvard Bus. Rev.*, 2020. [Online]. Available: https://hbr.org/2020/04/is-your-data-infrastructure-ready-for-ai

[113] K. Hazelwood et al., "Applied machine learning at Facebook: A datacenter infrastructure perspective," in *Proc. IEEE Int. Symp. High Perform. Comput. Archit.*, 2018, pp. 620–629.

[114] P. Canuma, "MLOps: What it is, why it matters, and how to implement it (from a data scientist perspective)," Blog post at Neptune, 2021. Accessed: Jun. 2021. [Online]. Available: https://neptune.ai/blog/mlops-what-it-is-why-it-matters-and-how-to-implement-it-from-a-data-scientist-perspective

[115] H. Harush, "Behind the buzzwords: How we build ML products at booking.com," Blog post at Medium, 2019. Accessed: Dec. 2020. [Online]. Available: https://medium.com/booking-product/behind-the-buzzwords-how-we-build-ml-products-at-booking-com-8140f8e47533

[116] J. Cogan, "The surprising truth about what it takes to build a machine learning product," Blog post at Medium, 2019. Accessed: Nov. 2020. [Online]. Available: https://medium.com/thelaunchpad/the-ml-surprise-f54706361a6c

[117] C. Mewald, "3 common problems with your machine learning product and how to fix them," Blog post at Medium, 2018. Accessed: Nov. 2020. [Online]. Available: https://medium.com/thelaunchpad/how-to-protect-your-machine-learning-product-from-time-adversaries-and-itself-ff07727d6712

[118] P. Ataee, "Build an ML product—4 mistakes to avoid," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/build-an-ml-product-4-mistakes-to-avoid-bce30d98bd24

[119] V. Karbhari, "MLOps: Real world model deployments," Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://medium.com/acing-ai/ml-ops-real-world-model-deployments-afab1fff9361

[120] ,"Machine learning pipeline: Architecture of ML platform in production," Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://altexsoft.medium.com/machine-learning-pipeline-architecture-of-ml-platform-in-production-70e38571a89a

[121] A. Soni, "MLOps: The epoch of productionizing ML models," Blog post at Medium, 2020. Accessed: Dec. 2020. [Online]. Available: https://medium.com/analytics-vidhya/mlops-the-epoch-of-productionizing-ml-models-4eec06d93623

[122] A. Dhinakaran, "Ml infrastructure tools for production," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. https://towardsdatascience.com/ml-infrastructure-tools-for-production-1b1871eecafb

[123] R. Sharma, "Hey Airbnb, do you want more dollar bills?" Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/crisp-dm-process-for-data-analysis-ed89db32cc88

[124] C. Mewald, "Your deep-learning-tools-for-enterprises startup will fail," Blog post at Medium, 2019. Accessed: Dec. 2020. [Online]. Available: https://medium.com/thelaunchpad/your-deep-learning-tools-for-enterprises-startup-will-fail-94fb70683834

[125] A. Munappy, J. Bosch, H. H. Olsson, A. Arpteg, and B. Brinne, "Data management challenges for deep learning," in *Proc. 45th Euromicro Conf. Softw. Eng. Adv. Appl.*, 2019, pp. 140–147.

[126] K. Pathak, "A look at machine learning system design," Blog post at Analytics Vidhya, 2021. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/01/a-look-at-machine-learning-system-design/

[127] , "Explainability + trust," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/explainability-trust/

[128] C. Chai, J. Wang, Y. Luo, Z. Niu, and G. Li, "Data management for machine learning: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4646–4667, May 2023.

[129] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A survey of devOps concepts and challenges," *ACM Comput. Surv.*, vol. 52, no. 6, pp. 1–35, 2019.

[130] A. Agrawal, J. Gans, and A. Goldfarb, "How to win with machine learning," *Harvard Bus. Rev.*, 2020. [Online]. Available: https://hbr.org/2020/09/how-to-win-with-machine-learning

[131] F. Candelon, R. C. di Carlo, and S. D. Mills, "AI-at-scale hinges on gaining a 'social license'," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://store.hbr.org/product/ai-at-scale-hinges-on-gaining-a-social-license/smr929?sku=SMR929-PDF-ENG

[132] K. Vaidhyanathan, A. Chandran, H. Muccini, and R. Roy, "Agile4MLS—Leveraging agile practices for developing machine learning-enabled systems: An industrial experience," *IEEE Softw.*, vol. 39, no. 6, pp. 43–50, Nov./Dec. 2022.

[133] S. Martínez-Fernández et al., "Software engineering for AI-based systems: A survey," *ACM Trans. Softw. Eng. Methodol.*, vol. 31, no. 2, pp. 1–59, 2022.

[134] P. Radanliev, D. De. Roure, R. Nicolescu, M. Huth, and O. Santos, "Digital twins: Artificial intelligence and the IoT cyber-physical systems in industry 4.0," *Int. J. Intell. Robot. Appl.*, vol. 6, no. 1, pp. 171–185, 2022.

[135] P. Radanliev, D. De Roure, R. Nicolescu, M. Huth, and O. Santos, "Artificial intelligence and the Internet of Things in industry 4.0," *CCF Trans. Pervasive Comput. Interact.*, vol. 3, pp. 329–338, 2021.

[136] B. Schneier, "Attacking machine learning systems," *Computer*, vol. 53, no. 5, pp. 78–80, 2020.

[137] D. Bourke, "2020 machine learning roadmap," Informations centered at GitHub, 2020. Accessed: Jun. 2021. [Online]. Available: https://github.com/mrdbourke/machine-learning-roadmap

[138] People + AI Guidebook, "Feedback + control," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/feedback-controls/

[139] P. Canuma, "Data science project management in 2021—The new guide for ML teams," Blog post at Neptune, 2021. Accessed: Jun. 2021. [Online]. Available: https://neptune.ai/blog/data-science-project-management-in-2021-the-new-guide-for-ml-teams

[140] T. Fountaine, B. McCarthy, and T. Saleh, "Getting AI to scale," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/05/getting-ai-to-scale

[141] C. L. Mat, "The 4 biggest problems I met while building machine learning products," Blog post at Medium, 2021. Accessed: Dec. 2022. [Online]. Available: https://medium.com/@cyrillemat/the-4-biggest-problems-i-met-while-building-machine-learning-products-83cad24f53eb

[142] R. N. Murty, et al., "Do you know how your teams get work done?," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/12/do-you-know-how-your-teams-get-work-done

[143] K. Sureshkumar, "Bayesian AB testing—part I—conversions," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/bayesian-ab-testing-part-i-conversions-ac2635f878ec

[144] H. Corona, "Why is re-training ML models important?," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/why-is-re-training-ml-models-important-a-product-managers-perspective-91d906d1ebe0

[145] D. Li, E. Hasanaj, and S. Li, "Baselines," Blog post at ML.CMU, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/3-baselines/

[146] A. Oleksyuk, "The perfect formula for fintech products: CX = ML UX," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/the-perfect-formula-for-fintech-products-cx-ml-ux-fc6ab01f136d

[147] D. Meister and R. Chandrasekhar, "Deploying artificial intelligence in new product development," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://store.hbr.org/product/mccormick-co-deploying-artificial-intelligence-in-new-product-development/w25509?sku=W25509-PDF-ENG

[148] K. Hume and M. E. Taylor, "Why AI that teaches itself to achieve a goal is the next big thing," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/04/why-ai-that-teaches-itself-to-achieve-a-goal-is-the-next-big-thing

[149] B. Ammanath and R. Blackman, "Everyone in your organization needs to understand AI ethics," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/07/everyone-in-your-organization-needs-to-understand-ai-ethics

[150] K. Dodson, "AI/ML: Shiny object or panacea?," Blog post at Cisco, 2018. Accessed: Jun. 2021. [Online]. Available: https://blogs.cisco.com/government/ai-ml-shiny-object-or-panacea

[151] F. Candelon, R. C. di Carlo, M. De Bondt, and T. Evgeniou, "AI regulation is coming," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/09/ai-regulation-is-coming

[152] D. Weinberger, "How machine learning pushes us to define fairness," *Harvard Bus. Rev.*, 2019. [Online]. Available: https://hbr.org/2019/11/how-machine-learning-pushes-us-to-define-fairness

[153] B. Ammanath, "Thinking through the ethics of new tech... before there's a problem," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/11/thinking-through-the-ethics-of-new-techbefore-theres-a-problem

[154] P. Meissner and C. Keding, "The human factor in AI-based decision-making," *Harvard Bus. Rev.*, 2021.

[155] R. Blackman, "A practical guide to building ethical AI," *Harvard Bus. Rev.*, 2021.

[156] P. Meissner and C. Keding, "AI should augment human intelligence, not replace it," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://hbr.org/2021/03/ai-should-augment-human-intelligence-not-replace-it

[157] J. Gans and K. Christensen, "Exploring the impact of artificial intelligence: Prediction vs judgment," *Harvard Bus. Rev.*, 2019. [Online]. Available: https://store.hbr.org/product/exploring-the-impact-of-artificial-intelligence-prediction-vs-judgment/rot376?sku=ROT376-PDF-ENG

[158] , "Errors + graceful failure," Blog post at Google.ai, 2019. Accessed: Jun. 2021. [Online]. Available: https://pair.withgoogle.com/chapter/errors-failing/

[159] L. E. Lwakatare, A. Raj, I. Crnkovic, J. Bosch, and H. H. Olsson, "Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions," *Inf. Softw. Technol.*, vol. 127, 2020, Art. no. 106368.

[160] I. Xiao, "Data product & economy 2.0," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/data-product-economy-2-0-a4263471fb56

[161] H. K. Jangir, "5 important things to keep in mind during data preprocessing! (specific to predictive models)," Blog post at Analytics Vidhya, 2021. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/01/5-important-things-to-keep-in-mind-during-data-preprocessing-specific-to-predictive-models/

[162] A. Malik, "Model risk management and the role of explainable models (with python code)," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/12/model-risk-management-and-the-role-of-explainable-modelswith-python-code/

[163] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying CRISP-DM process model," *Procedia Comput. Sci.*, vol. 181, pp. 526–534, 2021.

[164] A. Fard, A. Le, G. Larionov, W. Dhillon, and C. Bear, "Vertica-ML: Distributed machine learning in vertica database," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2020, pp. 755–768.

[165] M. N. Qureshi, "A quick guide to data science and machine learning," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/12/a-quick-guide-to-data-science-and-machine-learning/

[166] M. Liu, Z. Wang, Z. Gong, J. Yoon, and X. Wang, "Data exploration," Blog post at ML.CMU, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/2-data-exploration/

[167] F. Lucini, "The real deal about synthetic data," *Harvard Bus. Rev.*, 2021. [Online]. Available: https://store.hbr.org/product/the-real-deal-about-synthetic-data/smr933?sku=SMR933-PDF-ENG

[168] N. Lathia, "Machine learning for product managers," Blog post at Medium, 2017. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@neal_lathia/machine-learning-for-product-managers-ba9cf8724e57

[169] M. De-Arteaga, W. Herlands, D. B. Neill, and A. Dubrawski, "Machine learning for the developing world," *ACM Trans. Manage. Inf. Syst.*, vol. 9, no. 2, pp. 1–14, 2018.

[170] A. Ramani, "A product manager's guide to MLOps++," Blog post at Medium, 2021. Accessed: Dec. 2022. [Online]. Available: https://bytesforthought.medium.com/a-product-managers-guide-to-mlops-part-1-2-107fe4c9999a

[171] A. Agarwal, "Data validation in machine learning is imperative, not optional," Blog post at Analytics Vidhya, 2021. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/data-validation-in-machine-learning-is-imperative-not-optional/

[172] C. Fang, J. Ding, Q. Huang, T. Tong, and Y. Sun, "The overfitting iceberg," Blog post at ML.CMU, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/4-overfitting/

[173] R. Raghuwanshi, "Building ML products: Common pitfalls that product managers should avoid," Blog post at Medium, 2019. Accessed: Nov. 2020. [Online]. Available: https://medium.com/@rahulraghuwanshi_13455/building-ml-products-common-pitfalls-that-product-managers-should-avoid-14f393d1c250

[174] C. Alsisaria, "AutoML: Making AI more accessible to businesses," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/11/automl-making-ai-more-accessible-to-businesses/

[175] P. Pilotte, "Data pre processing: A crucial element of analytics driven embedded systems," Blog post at Analytics Vidhya, 2016. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2016/12/data-pre-processing-a-crucial-element-of-analytics-driven-embedded-systems/

[176] A. McKinney, "AI and the 4 keys to its success: Diving deeper," Blog post at Cisco, 2019. Accessed: Jun. 2021. [Online]. Available: https://blogs.cisco.com/government/ai-and-the-4-keys-to-its-success-diving-deeper

[177] M. Uddin, "Improving model management practices for machine learning teams," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/improving-model-management-practices-for-machine-learning-teams-3bec1dc40929

[178] A. Kumar, M. Boehm, and J. Yang, "Data management in machine learning: Challenges, techniques, and systems," in *Proc. ACM Int. Conf. Manage. Data*, 2017, pp. 1717–1722.

[179] M. Yeomans, "What every manager should know about machine learning," *Harvard Bus. Rev.*, vol. 93, no. 7, 2015. [Online]. Available: https://hbr.org/2015/07/what-every-manager-should-know-about-machine-learning

[180] S. Pandian, "Feature engineering (feature improvements scaling)," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/12/feature-engineering-feature-improvements-scaling/

[181] S. Jagannathan, "How to reduce annotation when evaluating AI systems," Blog post at Amazon Science, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.amazon.science/blog/how-to-reduce-annotation-when-evaluating-ai-systems

[182] S. Flender, "The four maturity levels of ML production systems," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/the-four-maturity-levels-of-ml-production-systems-94c4468e6b3

[183] P. Parchas, "How to reduce communication overhead for database queries by up to 97%," Blog post at Amazon Science, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.amazon.science/blog/how-to-reduce-communication-overhead-for-database-queries-by-up-to-97

[184] S. Gupta, "How to build data engineering pipelines at scale," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/how-to-build-data-engineering-pipelines-at-scale-6f4dd3746e7e

[185] H. B. Braiek and F. Khomh, "On testing machine learning programs," *J. Syst. Softw.*, vol. 164, 2020, Art. no. 110542.

[186] C. Han, "Speeding training of decision trees," Blog post at Amazon Science, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.amazon.science/blog/speeding-training-of-decision-trees

[187] P. Kalyan, "A comprehensive guide on feature engineering," Blog post at Analytics Vidhya, 2021. Accessed: Aug. 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-feature-engineering/

[188] B. Schreck, M. Kanter, K. Veeramachaneni, S. Vohra, and R. Prasad, "Getting value from machine learning isn't about fancier algorithms—It's about making it easier to use," *Harvard Bus. Rev.*, 2018. [Online]. Available: https://hbr.org/2018/03/getting-value-from-machine-learning-isnt-about-fancier-algorithms-its-about-making-it-easier-to-use

[189] M. Oleszak, "Don't let your model's quality drift away," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/dont-let-your-models-quality-drift-away-53d2f7899c09

[190] P. Dalmia, "An end-to-end guide to model explainability," Blog post at Analytics Vidhya. 2021. Accessed: Aug. 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/11/model-explainability/

[191] A. A. Awan, "A guide to machine learning pipelines and orchest," Blog post at Analytics Vidhya, 2021. Accessed: Aug. 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/10/a-guide-to-machine-learning-pipelines-and-orchest/

[192] M. W. Toffel, N. Epstein, K. Ferreira, and Y. Grushka-Cockayne, "Assessing prediction accuracy of machine learning models," *Harvard Bus. Rev.*, 2020. [Online]. Available: https://store.hbr.org/product/assessing-prediction-accuracy-of-machine-learning-models/621045?sku=621045-PDF-ENG

[193] A. Huang, J. Li, and N. Shankar, "Interpretability," Blog post at ML.CMU, 2020. Accessed: Jun. 2021. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/6-interpretability/

[194] A. Baku, "Effective AI infrastructure or why feature store is not enough," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/effective-ai-infrastructure-or-why-feature-store-is-not-enough-43bc2d803401

[195] A. Vianna, "How to implement data engineering in practice?" Blog post at Analytics Vidhya, 2021. Accessed: Aug. 2022. https://www.analyticsvidhya.com/blog/2021/12/how-to-implement-data-engineering-in-practice/

[196] Y. Haviv, "How to productize ML faster with MLOps automation," Blog post at Towards Data Science, 2020. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/how-to-productize-ml-faster-with-mlops-automation-d3f25caf38c4

[197] MANOABZ, "MLOps—The why and the what," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/11/mlops-the-why-and-the-what/

[198] S. Raheja, "A review of 2021 and trends in 2022—A technical overview of the data industry!," Blog post at Analytics Vidhya, 2021. Accessed: Aug.2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/12/a-review-of-2021-and-trends-in-2022-a-technical-overview-of-the-data-industry/

[199] Y. Haviv, "6 tips for MLOps acceleration & simplification," Blog post at Towards Data Science, 2021. Accessed: Nov. 2022. [Online]. Available: https://towardsdatascience.com/6-tips-for-mlops-acceleration-simplification-36539adab29b

[200] K. Jee, "Is data science dead in 10 years?," Blog post at Medium, 2021. Accessed: Dec. 2022. [Online]. Available: https://towardsdatascience.com/is-data-science-dead-in-10-years-3cde3963552

[201] M. Kumar, "A guide to deploying machine/deep learning model(s) in production," Blog post at Medium, 2018. Accessed: Dec. 2020. [Online]. Available: https://blog.usejournal.com/a-guide-to-deploying-machine-deep-learning-model-s-in-production-e497fd4b734a

[202] I. Rodrigues, "How to run a data science team: TDSP and crisp methodologies," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://medium.com/skim-technologies/how-to-run-a-data-science-team-tdsp-and-crisp-methodologies-13bec343b406

[203] R. S. Vuppuluri, "Yet another full stack data science project," Blog post at Towards Data Science, 2019. Accessed: Nov. 2020. [Online]. Available: https://towardsdatascience.com/yet-another-full-stack-data-science-project-a-crisp-dm-implementation-2943cae8da09

[204] L. Hardesty, "KDD: Where linear methods still have their place," Blog post at Amazon Science, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.amazon.science/blog/kdd-where-linear-methods-still-have-their-place

[205] D. Zheng, "How to train large graph neural networks efficiently," Blog post at Amazon Science, 2021. Accessed: Aug. 2022. [Online]. Available: https://www.amazon.science/blog/how-to-train-large-graph-neural-networks-efficiently

[206] P. Saikia, "The importance of data drift detection that data scientists do not know," Blog post at Analytics Vidhya, 2021. Accessed: Aug. 2022 [Online]. Available: https://www.analyticsvidhya.com/blog/2021/10/mlops-and-the-importance-of-data-drift-detection/

[207] A. Ponnada, "Machine learning model—Serverless deployment," Blog post at Analytics Vidhya, 2020. Accessed: Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/12/machine-learning-model-serverless-deployment/