Machine Learning Developments as Stimuli for Organizational Learning

Oliver A. Vetter
Technical University of Darmstadt, oliver.vetter@tu-darmstadt.de

Timo Sturm
Technical University of Darmstadt, timo.sturm@tu-darmstadt.de

Mariska Fecho
Technical University of Darmstadt, mariska.fecho@tu-darmstadt.de

Peter Buxmann
Technical University of Darmstadt, buxmann@is.tu-darmstadt.de

Machine Learning Developments as Stimuli for Organizational Learning

Completed Research Paper

Oliver A. Vetter[1]
Technical University of Darmstadt
Hochschulstraße 1, 64289 Darmstadt
oliver.vetter@tu-darmstadt.de

Timo Sturm[1]
Technical University of Darmstadt
Hochschulstraße 1, 64289 Darmstadt
timo.sturm@tu-darmstadt.de

Mariska Fecho
Technical University of Darmstadt
Hochschulstraße 1, 64289 Darmstadt
mariska.fecho@tu-darmstadt.de

Peter Buxmann
Technical University of Darmstadt
Hochschulstraße 1, 64289 Darmstadt
peter.buxmann@tu-darmstadt.de

Abstract

Organizational learning is a fundamental process that defines organizational behavior and thereby strongly influences organizational performance. As organizations increasingly adopt machine learning (ML) systems in their routines, the need to illuminate the impact of learning machines on organizational learning processes becomes increasingly urgent. In particular, due to their highly interdisciplinary and collaborative nature, ML developments—as organizations' activities aimed at creating productively usable ML systems—may hereby represent an important but not yet well understood mechanism for fostering organizational learning. To explore how ML developments affect organizational learning processes, we interviewed 42 experts who are frequently involved in ML developments. Our findings suggest that ML developments can enhance organizational learning by stimulating a variety of organizational learning processes that generate a wealth of explicit and tacit knowledge in organizations.

Keywords: Organizational learning, machine learning, development, knowledge, tacit, explicit

Introduction

Organizational learning lies at the heart of organizational behavior because it represents the process by which organizations continually (re)define their norms, innovations, and routines (e.g., Argote & Miron-Spektor, 2011; Levitt & March, 1988). Learning thereby enables organizations to effectively adapt to their environment, which is critical to their success and, if done wrong, can even jeopardize the organizations' survival (e.g., Argote et al., 2021; Huber, 1991; March, 1991). Since organizations cannot learn on their own, they must rely on the learning of their members (e.g., Fang et al., 2010; March, 1991). Therefore, facilitating and effectively coordinating the individual learning processes and interactions of its members represent the key issue behind effective organizational learning, which has already spurred decades of research (great overviews exist such as, e.g., Argote et al., 2021; Argote & Miron-Spektor, 2011; Fiol & Lyles, 1985; Huber, 1991). However, as human members have traditionally been the ones who learn in organizations, the limitations of human cognition have complicated organizational learning since its very beginnings (e.g., Levinthal & March, 1993; March, 2006, 2010; Simon, 1991).

[1] NOTE: Both authors contributed equally to this research.

Hoping to overcome the limitations of human cognition, organizations are increasingly recognizing the great potential of information systems (IS) based on machine learning (ML) algorithms (e.g., Berente et al., 2021; Benbya et al., 2021). Such ML systems can learn autonomously by inferring patterns from data to create models for guiding behavior (e.g., Mitchell, 1997; Russel & Norvig, 2021). Because of their ability to learn, recent research now also considers ML systems as a new type of organizational learner besides humans; that is, not just as another tool that only supports human learning (like traditional non-ML IS; e.g., Argote et al., 2021; Alavi & Leidner, 2001, Kane & Alavi, 2007), but rather as active learners that can contribute their own knowledge to organizational learning (e.g., Argote et al., 2021; Balasubramanian et al., 2022; Sturm et al., 2021a). Indeed, the great computational power of ML systems, which have already beaten the best human Go player, surpassed human capabilities in object recognition, and recently begun to revolutionize the way we write text, makes them increasingly appear as a panacea for improving organizational efficiency and effectiveness (e.g., Jordan & Mitchell, 2015; Lindebaum et al., 2020).

Despite this great potential, many organizations struggle to create productive ML systems. As a result, many organizations are afraid to waste their scarce resources and reduce their funding for ML developments (e.g., Ransbotham et al., 2020; Sculley et al., 2015), which are typically project-led efforts focused on developing ML systems for productive use (e.g., Amershi et al., 2019; Studer et al., 2021). While this may seem like an appropriate risk mitigation strategy at first glance, it may also become a fatal fallacy, as such ML developments may act as a powerful form of organizational learning: from data selection to ML model evaluation to knowledge sharing with ML systems (e.g., Amershi et al., 2019; Wirth & Hipp, 2000), the development and use of ML systems involves an interdisciplinary learning process among an organization's domain experts, data scientists, and/or ML algorithms that would not otherwise occur (e.g., knowledge sharing sessions between data scientists and engineers to develop a predictive maintenance solution). Since ML developments require reflection on existing knowledge (e.g., through collective data exploration) and can enable new knowledge creation beyond the mere development of ML systems (e.g., insights into flaws in routines), they may have a significant impact on the use, extension, and retention of an organization's knowledge. ML developments may therefore play a crucial role in facilitating and stimulating organizational learning. The discontinuity of ML developments may thus lead to a critical competitive disadvantage in the long run—even if an organization's ML developments frequently fail to yield productive ML systems.

So far, however, the role of ML developments for organizational learning has been largely neglected by research. Unfortunately, research on the general impact of IS on organizational learning can only be of limited help, as existing studies primarily focus on the impact of IS use rather than IS development (e.g., Alavi & Leidner, 2001; Argote et al., 2021; Argote & Miron-Spektor, 2011; Kane & Alavi, 2007) and naturally neglect the particularities of ML systems (e.g., involvement of data scientists and system functionality being defined primarily by data analysis rather than human-defined rules; Amershi et al., 2019; Brynjolfsson & Mitchell, 2017; Sturm et al. 2021). Only recently have a few scholars begun to explore the impact of productive ML systems as a new type of organizational learner on organizational learning (e.g., Argote et al., 2021; Balasubramanian et al., 2022; Lyytinen et al., 2021; Ransbotham et al., 2020; Sturm et al., 2021a, 2021b). Yet, research that takes a broader view to include the impact of *human* learning processes that occur within ML developments on organizational learning is still lacking, leaving organizations without clear guidance. To help organizations better manage and grasp the value of their ML developments, we aim to explore the impact of ML developments on different organizational learning processes. We thus ask the following research question (RQ): *How do ML developments affect organizational learning processes?*

To answer our RQ, we adopt a qualitative research approach that allows us to explore and synthesize the experiences of 42 experts who have been frequently involved in ML developments. Our findings suggest that ML developments can indeed contribute value to organizational learning. We further find that these contributions involve different learning processes depending on the ML development phase. Our result is a framework for how ML developments can stimulate different types of organizational learning processes.

Theoretical Background

We first introduce organizational learning, with a particular focus on Nonaka's (1994) seminal work on the knowledge creation spiral. We then turn to the foundations of ML and the typical processes within ML developments that we revisit as a form of problem solving. Table 1 summarizes the core concepts that we introduce in both subsections and which will act as a theoretical structure for our subsequent analysis. We conclude by crystallizing the need to revisit organizational learning in the context of ML developments.

Source	Concept	Definition	Examples
Nonaka (1994)	Knowledge types	**Explicit knowledge**: Knowledge that is codified in some form and easy to articulate.	*Manuals, standard operating procedures, and patents*
		Tacit knowledge: Knowledge that is deeply rooted in personal experience and difficult to articulate.	*Riding a bike, playing an instrument, and dancing a choreography*
	Learning processes	**Socialization (tacit to tacit)**: Integrating tacit knowledge by creating shared experiences.	*Apprenticeship, mentoring, and immersion in a community of practice*
		Externalization (tacit to explicit): Codifying tacit knowledge into explicit knowledge.	*Capture personal experiences as a metaphor, a concept, or a story*
		Combination (explicit to explicit): Integrating different explicit knowledge by combining documented concepts.	*Integrate rule sets, documents, and/or frameworks*
		Internalization (explicit to tacit): Creating tacit knowledge by gaining personal experience with explicit knowledge.	*Apply textbook knowledge about skating, singing, and cooking to gain personal experience with it*
Basadur et al. (1982)	Problem-solving processes	**Problem finding:** Recognize and construct relevant problems.	*Identify and describe bottlenecks and decision-making flaws*
		Problem solving: Search for adequate solutions by exploring potential solutions for given problems.	*Define quality requirements, test and compare solution candidates*
		Solution implementation: Integrate selected solutions into organizational processes.	*Provide additional tools to accelerate a time-consuming process*

Table 1. Core Knowledge Types and Core Learning and Problem-Solving Processes

Organizational Learning

Organizational learning is the process of gathering experience in some organizational context and deriving knowledge from that experience to guide future actions (e.g., Argote & Miron-Spektor, 2011; Levitt & March, 1988; March, 2010; Nonaka, 1994). Experience thereby denotes one's own or others' unit of task performance (e.g., making decisions, observing others performing routines). Learners' experiences are thus assumed to primarily contain recollections of chosen actions and their consequences, enriched with information about the context in which the actions took place (e.g., Argote et al., 2021; Argote & Miron-Spektor, 2011). Learners then learn from the available experience by inferring conclusions from it, attempting to generalize and reconcile the collected knowledge (e.g., decision rules, approaches to performing routines) with existing knowledge (e.g., Argote & Miron-Spektor, 2011; March, 2010). Thereby, learners can learn from others' experiences either directly (e.g., by drawing conclusions from observing others' actions) or indirectly by integrating others' knowledge with their own knowledge (e.g., by integrating their own and others' decision rules; e.g., Argote & Miron-Spektor, 2011; Nonaka, 1994).

To further nuance organizational learning processes, Nonaka (1994) famously introduced the now seminal theory of the knowledge creation spiral. In his theory, Nonaka (1994) first introduces two core knowledge types (i.e., *explicit* and *tacit* knowledge) and then outlines four core learning processes that focus on integrating and converting between the two knowledge types (see also Table 1 for an overview). First, ***explicit knowledge*** refers to knowledge that is codified or documented in some form, such as written or electronic documents, databases, or other tangible formats. Explicit knowledge can be easily articulated, shared, and accessed by others. Examples include textbooks, technical manuals, standard operating procedures, patents, and scientific articles. Second, ***tacit knowledge*** refers to knowledge that is difficult to articulate, codify, or transfer to others. This type of knowledge is often deeply rooted in an individual's personal experience, intuition, and insights, and it may be difficult to formalize or explain in words. Tacit knowledge is typically acquired through personal experience, observation, and informal learning. For instance, tacit knowledge are skills like riding a bike, playing an instrument, and dancing a choreography as they involve a complex interplay of physical and mental abilities that are difficult to describe in words and thus hard to teach explicitly through written or spoken instructions.

To optimize knowledge creation, Nonaka (1994) argues that organizations must enable a continuous conversion between both types of knowledge. To do so, Nonaka (1994) identified four iteratively connected processes: First, ***socialization*** involves the sharing and integration of tacit knowledge by creating shared experiences through social interaction, observation, and emulation. For instance, socialization includes learning from others through apprenticeship, mentoring, and immersion in a community of practice. By developing a shared understanding of a particular domain within a specific context, humans gather personal experience that helps them approximate others' tacit knowledge. Second, ***externalization*** focuses on the transformation of tacit knowledge into explicit knowledge. By doing so, externalization can help facilitate the communication of tacit knowledge to some extent. For instance, this includes capturing personal experiences as a metaphor, a concept, or a story. Third, ***combination*** focuses on creating new explicit knowledge by integrating different explicit knowledge that already exists. Here, individuals synthesize different forms of explicit knowledge, such as rule sets, documents, and frameworks. Thereby, combination allows individuals to build on established knowledge to create new insights and perspectives. Fourth, ***internalization*** involves the transformation of explicit knowledge into tacit knowledge. By applying explicit knowledge to the particularities of specific contexts, individuals can gain a deeper understanding of the knowledge and its applications, and thereby develop new tacit knowledge based on the gathered experiences. For instance, a skateboarder can internalize textbook knowledge of a certain trick, by enriching the knowledge with their own related experience by trying the trick themselves.

Nonaka (1994) theorizes the spiral of knowledge creation as an iterative process that continuously traverses all four learning processes, shifting between explicit and tacit knowledge. Because of the spiral's iterative nature, effective organizational learning requires an organization to facilitate each of the four learning processes to enable well-functioning, mutually stimulating knowledge creation.

Machine Learning Development

The approach behind modern artificial intelligence (AI) that has driven recent breakthroughs (e.g., ChatGPT or AlphaGo) is the use of ML algorithms (e.g., Berente et al., 2021; Brynjolfsson & Mitchell, 2017). ML algorithms allow ISs to derive patterns from data to create ML models that are then used to solve given problems (e.g., deriving a rationale for how to grant a loan; e.g., Mitchell, 1997; Russel & Norvig, 2021). In doing so, ML resembles organizational learning: ML systems use experience (i.e., data capturing units of task performance) to infer conclusions from it (i.e., models of how to perform a task), thereby attempting to generalize and reconcile the contained knowledge (e.g., Sturm et al., 2021a). In contrast to traditional non-ML IS that were only able to support human learning (e.g., emails, Zoom, or repositories supporting human knowledge exchange; e.g., Alavi & Leidner, 2001; Kane & Alavi, 2007), this enables ML systems to learn autonomously and contribute their own knowledge to organizations' stock of knowledge (e.g., Argote et al., 2021; Sturm et al., 2021a). As a result, ML systems can be viewed as a new type of organizational learner besides humans (e.g., Balasubramanian et al., 2022; Ransbotham et al., 2020; Sturm et al., 2021a).

ML systems are realized through ML developments, which are project-led efforts to develop and deploy ML systems productively (e.g., Amershi et al., 2019; Studer et al., 2021). Such ML developments are typically a highly iterative and interdisciplinary practice involving a diverse set of collaborating domain and technical experts (e.g., Amershi et al., 2019; Wirth & Hipp, 2000). For example, developing an ML system for predicting wind turbine failures requires domain experts with knowledge of wind turbine behavior to help define the problem, and data scientists to perform analyses and translate them into a suitable solution, which must then be evaluated again by the domain experts. To coordinate this process, numerous process frameworks have been proposed for ML developments (e.g., CRISP-DM, SEMMA, and KDD) that are widely used in practice (e.g., Amershi et al., 2019; Azevedo & Santos, 2008; Martinez et al., 2021; Martínez-Plumed et al., 2021). While these frameworks differ in their specific process steps, they share a common ground of key phases that resemble established phases of problem-solving processes as famously conceptualized by Basadur et al. (1982). Following this rationale, we can revisit these key phases for ML development contexts:

ML developments typically begin with a ***problem finding*** phase, which focuses on articulating a relevant problem with data that a planned ML system is intended to solve. First, this requires experts to identify problems that are organizationally meaningful and adequate for ML systems to solve. This includes activities such as exploring business domains and available data to identify bottlenecks and flaws in organizational processes or practices. Second, once problems have been identified, experts need to create a concise and realistic representation of the identified problems. This includes activities such as exploring the availability of data, selecting a data sample that is a representative and comprehensive collection of exemplary problem instances, and preparing data to ensure high data quality (e.g., accuracy, completeness, timeliness, and consistency) and to develop additional variables to describe problem instances more holistically. The second phase of ML developments typically focuses on ***problem solving***, which involves the search for ML models that adequately suit the identified problems. This first requires the organization to develop a shared understanding of what an appropriate ML model should entail. This includes activities such as defining baselines for evaluation metrics for when an ML model is good enough to be used, and clarifying transparency requirements for ML models (e.g., requiring highly transparent models such as decision trees). Second, once the requirements for appropriate ML models have been defined, experts search for possible ML models by creating and evaluating a variety of ML models, and identify the seemingly best ML model. This includes activities such as selecting ML algorithms, parameterizing ML algorithms, and training, testing, and comparing prototypical ML models. Lastly, the third phase focuses on ***solution implementation***, which aims to integrate ML systems into organizational processes. This typically involves redesigning established processes to include ML systems and to design and sustain their emerging interplay with humans. For instance, this requires organizations to rethink how inputs and outputs of ML systems can be integrated to ensure effective workflows and how to adapt the role of involved humans.

As ML developments are therefore a highly interdisciplinary process that involves collective reflection on given problems, potential solutions, and the integration of knowledge between multiple human experts (and ML systems), organizations' ML endeavors may not only impact organizational learning through productively used ML systems that now learn side-by-side with humans but also by the additional human learning processes that are stimulated by the processes entailed in ML developments.

The Need to Revisit Organizational Learning in the Context of ML Developments

For decades, researchers have analyzed the processes of *human*-driven organizational learning and how organizations can effectively coordinate these processes to improve organizational performance (e.g., Argote et al., 2021; Argote & Miron-Spektor, 2011; Fiol & Lyles, 1985; Huber, 1991). However, with their ability to learn, ML systems increasingly participate as a new type of learner in organizational learning alongside humans (e.g., Ransbotham et al., 2020; Sturm et al., 2021a). While research has recently recognized the importance of understanding ML's impact on organizational learning (e.g., Argote et al., 2021; Balasubramanian et al., 2022; Berente et al. 2021; Sturm et al., 2021a), the few studies that exist focus only on the impact of ML systems that are already in *productive use*. While these studies provide crucial insights on how to effectively manage the resulting learning dynamics between humans and ML systems, this focus neglects the potential impact of the underlying ML development activities required to enable such productive ML systems in the first place. This is problematic because research that overlooks ML developments may thereby neglect a novel context of analytically-driven interactions between domain and technical human experts that are likely to offer new opportunities for stimulating organizational learning processes (e.g., human experts sharing their domain knowledge to prepare data for training and evaluating a planned ML system). Moreover, if organizations frequently fail to produce productively usable ML systems, they run the risk of misjudging the impact of reducing their ML developments, which may help them save scarce resources in the short term, but may become a fatal long-term fallacy if they thereby inhibit valuable organizational learning processes. While existing research emphasizes the need for further analysis of the impact of ML systems on organizational learning due to such consequences (e.g., Argote et al., 2021; Sturm et al., 2021a), research that also considers the impact of the preceding ML developments remains non-existent. As with existing ML research, research on the general impact of IS on organizational learning focuses mainly on IS use rather than IS development (e.g., Alavi & Leidner, 2001; Argote et al., 2021; Argote & Miron-Spektor, 2011; Kane & Alavi, 2007), and inherently neglects the particularities of ML developments (e.g., Amershi et al., 2019; Brynjolfsson & Mitchell, 2017; Sturm et al. 2021)—also leaving us with limited help in unpacking the impact of ML developments. As a result, the current discussion runs the risk of being too narrowly focused, which can lead to ill-informed decisions for organizations. Hoping to contribute to broadening the perspective of the current discussion, we now turn to our study to analyze how ML development activities contain opportunities to stimulate different learning processes and thus can serve as important mechanisms to improve the long-term performance of organizations.

Qualitative Research Methodology

Due to the lack of research on organizational learning in ML developments, we pursued a qualitative research approach through interviews with professionals from various industries. Expert interviews are one of the most important data collection tools in research contexts that lack sufficient empirical evidence (Myers & Newman, 2007), allowing us to examine ML developments' impacts on organizational learning in a wide variety of contexts. Our goal is to develop a theoretical foundation for organizational learning processes involving humans and ML systems that occur during the different ML development phases.

We conducted 42 in-depth semi-structured interviews with experts who are frequently involved in ML developments. Each interview was conducted online and lasted 55 minutes on average. The experts were recruited from our personal and professional networks, primarily through LinkedIn (Butts et al., 2015). To broaden the scope of our analyses and develop a more general theory, we interviewed experts from a variety of different industries between March 2021 and July 2022 (Davidson & Martison, 2016). In total, our experts' experiences span 15 industries, with software (23.8%), manufacturing (11.9%), and telecommunication (11.9%) being the most represented. Most experts work for inter- or multinational organizations. To capture diverse perspectives on ML developments, we included both technology- and domain-oriented experts. Having worked on various ML developments over several years, the experts can offer a deep and diverse set of experiences. Table 2 provides an overview of the interviewed experts.

ID	Position	Sex	Experience	Industry	ID	Position	Sex	Experience	Industry
I1	Data scientist	m	3 yrs.	Aviation	I22	Data scientist	m	12 yrs.	Trade
I2	Data scientist	w	6 yrs.	Software	I23	Data scientist	m	6 yrs.	Tourism
I3	Data scientist	m	8 yrs.	Software	I24	Data scientist	m	3 yrs.	Technical Testing
I4	Domain expert	m	5 yrs.	Software	I25	Data scientist	m	4 yrs.	Automotive
I5	Data scientist	w	6 yrs.	Healthcare	I26	Data scientist	w	8 yrs.	Software
I6	Domain expert	m	9 yrs.	Telecommunication	I27	Domain expert	w	6 yrs.	IT Consulting
I7	Domain expert	m	12 yrs.	Software	I28	Manager	m	6 yrs.	Telecommunication
I8	Data scientist	m	8 yrs.	E-Commerce	I29	Manager	m	7 yrs.	IT Consulting
I9	Domain expert	w	5 yrs.	Energy	I30	Data scientist	w	3 yrs.	Telecommunication
I10	Domain expert	m	6 yrs.	Automotive	I31	Data scientist	m	7 yrs.	Manufacturing
I11	Data scientist	m	7 yrs.	Software	I32	Manager	m	3 yrs.	Manufacturing
I12	Data scientist	m	5 yrs.	Automotive	I33	Data scientist	m	6 yrs.	Research
I13	Domain expert	m	4 yrs.	Automotive	I34	Domain expert	m	3 yrs.	Software
I14	Data scientist	m	3 yrs.	Healthcare	I35	Data scientist	m	3 yrs.	IT Consulting
I15	Domain expert	m	5 yrs.	Manufacturing	I36	Manager	w	5 yrs.	Infrastructure
I16	Data scientist	m	9 yrs.	Market Research	I37	Data scientist	m	2 yrs.	Manufacturing
I17	Data scientist	w	2 yrs.	Healthcare	I38	Manager	m	6 yrs.	Software
I18	Domain expert	m	4 yrs.	IT Consulting	I39	Manager	w	7 yrs.	Aviation
I19	Data scientist	m	7 yrs.	Manufacturing	I40	Domain expert	w	7 yrs.	Healthcare
I20	Data scientist	m	8 yrs.	Telecommunication	I41	Manager	m	5 yrs.	Software
I21	Data scientist	w	11 yrs.	Software	I42	Domain expert	m	10 yrs.	Telecommunication

Table 2. Overview of the Interviewed Experts

At the beginning of each interview, the interviewees were introduced to the topic and our RQ. The interview guide consisted of a series of open-ended questions and was divided into four sections: The first section aimed to familiarize the interviewees with the interview situation and to develop an understanding of the interviewees' general expertise (i.e., descriptions of participants' current position, past ML development experience, and current ML development involvement). The subsequent three sections then focused on the three ML development key phases: Problem finding (i.e., how the interviewees identify appropriate problems and prepare related data), problem solving (i.e., how the interviewees train and evaluate ML models), and solution implementation (i.e., how ML systems are used productively in their organization). We used the guide as a basis for structuring our interviews according to the prepared questions, while also allowing for improvisation and spontaneous questions based on the particular course of the interview (Myers & Newman, 2007). After mutual consent, the interviews were recorded and transcribed to analyze the participants' responses. For interview I13, we were only able to take notes instead of making a recording.

As mentioned above, our goal is to identify learning processes that occur within ML developments. We analyzed the interview data by following the directed content analysis, which is appropriate for validating and extending existing theories (Hsieh & Shannon, 2005). An iterative multi-cycle, multi-researcher coding process with the following coding strategies was performed (Saldaña, 2015): First, we employed attribute coding to select descriptive and essential information about participants' characteristics. Second, we then utilized hypothesis coding to create primary codes according to the related literature of organizational learning. We used these primary codes, based on previous findings about human learning in organizations, as a structural framework to categorize the codes specific to the ML context that emerged in the following steps, including the codes *externalization*, *combination*, *internalization,* and *socialization* along each problem-solving phase (i.e., *problem finding*, *problem solving*, and *solution implementation*) of ML development. Third, we applied descriptive coding to uncover novel insights into learning processes during ML developments, which produced numerous subcodes for each primary code. Fourth, to make sense of the vast amount of codes and to remove redundant and irrelevant codes, we used pattern coding to integrate the identified codes into emergent themes that represent major subprocesses of the learning processes in the problem-solving phases (e.g., "*tacit knowledge about relevant variables becoming more structured and externalized when reflecting on possible data for a given problem*" was a code categorized into the theme "*domain experts externalize their problem understandings*" of the *externalization* processes in the *problem finding* phase). Each of us conducted these steps separately in each iteration, after which we intensively discussed our findings and synthesized commonalities into a shared code system to gradually develop a consensus on the codes. As part of the iterative process, we continued to collect new data to answer questions that arose during coding until no new insights emerged. As we observed no new insights in the final interviews, we noticed that we had reached theoretical saturation and stopped interviewing after the 42nd interview (Flick, 2004). Finally, we formulated summaries (shown in Table 3) of the main themes and selected quotes from our experts that best represent the corresponding codes, which we describe below.

Results

Combining the four processes of Nonaka's (1994) knowledge creation spiral with Basadur et al.'s (1982) problem-solving phases provides us with a fruitful structure for exploring the learning processes that occur within ML developments. Table 3 summarizes the learning processes we have identified as occurring in the three phases of ML development, which we now describe below.

	Problem finding Main stimulus: Data scientists Main learning focus: Problems	**Problem solving** Main stimulus: Data scientists Main learning focus: Solutions	**Solution implementation** Main stimulus: ML systems Main learning focus: Solutions
Externalization (tacit → explicit)	Domain experts externalize their problem understandings.	Domain experts externalize their understanding of an adequate solution.	Domain experts' solutions get externalized by ML systems.
Combination (explicit → explicit)	Domain experts compare actual observed and expected problem behavior.	Domain experts integrate insights of candidate solutions with their solutions.	Domain experts integrate their solutions with those suggested by ML systems.
Internalization (explicit → tacit)	Domain experts experiment with newly discovered types of problem instances.	Domain experts experiment with solutions suggested by data scientists.	Domain experts experiment with solutions suggested by ML systems.
Socialization (tacit → tacit)	Domain experts share experiences and resolve conflicting problem understandings.	Domain experts exchange their solutions and resolve uncovered conflicts.	Domain experts share experiences and resolve conflicting solutions with ML systems.

Table 3. Identified Learning Sub-Processes in ML Developments

Problem finding

As detailed above, the problem finding phase of ML developments focuses on recognizing problems and constructing representations of those problems in data. During our interviews, it became apparent that data scientists serve as the main stimulus for the various learning processes that take place in this phase as they approach domain experts with derived data insights in different ways, which we describe below.

Externalization: Our experts emphasized that at the beginning of ML developments, domain experts (experts with deep knowledge of the focused problem domain but without deep data science knowledge; e.g., bankers, engineers) and data scientists (experts with deep data science knowledge but without deep domain knowledge) typically begin by participating in knowledge sharing sessions. While these sessions aim to help the data scientists gain a rough understanding of the problem domain, our interviews revealed that these sessions involve more than just transferring the domain experts' understanding of the problem to the data scientists. By having to articulate their often implicit understanding of problems to data scientists, who are typically laypeople in the given domain, domain experts are forced to reflect on their tacit knowledge of a problem and explain it in a simple and transparent way along explicit variables that can potentially be reflected in the data—an exercise that domain experts often only rarely have to perform in their day-to-day work in the domain, but which helps them better clarify their understanding to themselves and make it readily available to others (since their normal practice requires them only to act on the problem, but not to clearly explain their rationale for doing so):

> *"We often start with the data collection and then a bunch of us get together, look at some documents, including data scientists, product managers, and some other people, to see what's in there. Then we talk to the [domain experts] to learn what they need, what they want. And then we come together and decide what the desired data collection looks like. [...] And through that process, we often learn a lot more about the problem itself."* (I34)

The experts further emphasized that these articulations are then carefully documented during the sessions because they are needed for subsequent data preparation:

> *"What exactly is the problem we focus on? What is the purpose of the project? What do we want to investigate in the first place? What do we want to predict somehow? That's exactly what it means to initially try to understand the processes and to understand the problem correctly, and then to outline with the domain experts what kind of data we need."* (I17)

> *"You first have to find out what problem is behind the data, so creating something like a documentation or something like that is an issue."* (I6)

In this way, ML developments can help organizations clarify, express, and document critical tacit knowledge on the respective problem domain that has remained exclusive to individual experts, thereby making it accessible to others in the form of explicit knowledge.

Combination: Our experts stated that, based on the understanding gained in the initial knowledge sharing sessions, data scientists then typically begin to explore and prepare the available data to progressively understand the data and improve data quality. In doing so, our experts explained that data scientists engage in an ongoing exchange with domain experts to clarify their understanding of why certain data characteristics may occur (e.g., data distributions, correlations, outliers). To do this, data scientists typically confront domain experts with their objective data descriptions, and possibly their own "lay" hypotheses, and ask for an explanation. This requires domain experts to reflect on their expected and the actual explicit problem behavior observed in the data and formulate a plausible explanation. To do this, domain experts often first explicate their reasoning of why certain problem behavior (and thus data characteristics) can occur, and then use these explanations to challenge the actual observed problem behavior. To this end, this process can help domain experts evaluate their understanding of the problem behavior: If their expected and the actual observed problem behaviors match, then this practice reinforces their correct understanding. If they differ, however, domain experts and data scientists typically engage in a mutual process of sensemaking, in which they gradually combine the domain experts' explanations and the explicit results of the data scientists' data analyses to extend or revise the domain experts' understanding of the problem domain. In particular, an important case that can stimulate this process is when data scientists identify outliers in the data, forcing domain experts to explain and reflect on (ab)normal problem behavior that may have remained hidden to them:

> *"When you're monitoring an IoT device, you may realize at some point that some operations can be performed better or more efficiently and then you communicate this to [the domain experts], who have often not thought about it this way." (I24)*

> *"We would preprocess the data and sit with the domain experts to show them what we are able to see in the data, and then ask them for explaining something that seems off. In doing so, we were sometimes able to detect that some of the regular maintenance they did was not useful. So they would learn a lot of things about the practices, something that they do, the norms, unexpectedly." (I5)*

In this way, ML developments can help organizations evaluate, revise, and extend explicit knowledge by combining explicated domain knowledge with data insights.

Internalization: Our experts further described that after being confronted with explicit data insights, domain experts often engage in further investigation to better understand problem instances they observed in the data but were unfamiliar with why they occur and how they behave—especially when the insights reflect anomalies in their domain. In doing so, domain experts use the available insights to guide their own experimentation with previously unfamiliar problems to improve their understanding of them, unpacking the conditions under which they may occur and the consequences of dealing with them in different ways. In the process, domain experts gain new experience with novel problems observed in others' experiences reflected in the data, which they can use to enrich their own experience and ultimately revise their established practices and develop new approaches to account for the novel problem instances (e.g., considering special cases in their day-to-day decision making that they may have overlooked so far):

> "*We actually found a few things from the data and some of it just by looking at it, so it was just the analysis—and maybe the visualization, that was also very important. [...] And then, on the [domain expert] side, people said: oh, that's interesting, we have to take a closer look at that.*" (I4)

> *"Through this data preparation, we can actually see where things are not working so well, where the data pipelines are incomplete, where business logic is being applied that should not actually be applied, where quality issues occur. That helps us data scientists, but on the other hand [...], of course, we feed this back to the responsible domain experts by reaching out to them like: 'In this step, the process is not working as described. You should investigate in the process why this is the case.'"* (I40)

In this way, ML developments can help organizations foster the exploration of unusual experiences to diversify individuals' tacit knowledge.

Socialization: Finally, our experts also described how collective data analysis can foster socialization of experts into each other's experiences and underlying rationales for performing routines and making decisions. The experts stated that, to ground their assumptions well, data scientists often reach out to more than one domain expert to gather multiple opinions about certain ambiguities in the data, with the goal of assessing whether there exists a consensus among domain experts. In doing so, data scientists confront multiple domain experts with data insights at the same time, which often stimulates discussions among the domain experts about the correctness of actual practices that can be observed in the data. This often reveals existing ambiguities and, in particular, inconsistencies between the approaches of domain experts—in the form of so-called "aha moments"—which in turn enable valuable consensus-building processes in ongoing discussions and evaluations (e.g., agreeing on compromises or defining exceptions). Especially when the domain experts come from different departments and thus base their reasoning on different organizational contexts and experiences, the stimulated socialization can help domain experts "look outside their ordinary box" and thereby reflect, revise, and extend their accumulated expertise:

> "*Or what we also have quite often, which is always very, very exciting, when we have two case workers in a room and we look at an outlier, then one says, 'Yes, yes, it's clear, decision A was correct'. And the other one looks at the one with big eyes and says, 'Nah, I would definitely go for B.'*" (I9)

In this way, ML developments can help organizations share and resolve complementary and conflicting tacit knowledge about the collectively faced problem domain.

Problem solving

The problem solving phase aims to build the ML systems that can best solve the problems represented in the prepared data. To this end, data scientists typically perform the training of ML models to propose candidate solutions, while domain experts are heavily involved in their evaluation to identify the seemingly best candidate. As in the problem finding phase, our interviews suggest that data scientists act as the main stimulus of involved organizational learning processes: Data scientists repeatedly approach domain experts to let them evaluate their derived candidates of ML models, which stimulates different learning processes that help domain experts gradually improve their understanding of adequate solutions, as follows.

Externalization: Since data scientists lack a deep insight into the domain, our experts emphasize that domain experts are typically urged to articulate their perception of an adequate solution in an easy-to-understand manner to data scientists, allowing them to define and document the requirements that an ML system must meet to adequately solve a given problem. In this way, domain experts describe the requirements that seem most relevant to them, while data scientists act as a kind of translator between domain experts and ML algorithms (e.g., translating their understanding of described solution requirements into available data variables and quantifiable metrics that can be interpreted by ML algorithms). In addition, our experts highlight that data scientists expect domain experts to identify and correct anomalous learning by the ML systems when evaluating them, which typically requires them to justify and explain why they believe certain behavior of an ML model is inadequate for solving a problem:

> "*You start with a very rudimentary model, maybe basic distinctions it can do. For example, for some sort of vision model, we started off with some basic classes [...] which you then show to the business to get their buy-in. Then, how do you need to fine tune it? What kind of granularity do you eventually need? This is where you need to involve the domain experts.*" (I22)

> *"Simply dealing with the situation or with the systems leads to the emergence of knowledge, i.e. a great deal of implicit knowledge is made explicit. [...] Often you have specialist departments that have been doing this and that all their lives and for them it is completely intuitive, so to speak, and they implicitly know how to do it. If you then build software from it or deal with the subject matter, then you have to do it all explicitly and it is discussed. By doing so, the knowledge becomes explicit and then usually ends up in some documentation."* (I29)

In this way, ML developments can help organizations externalize and document tacit knowledge about existing perceptions of adequate problem solutions.

Combination: Our experts described that, when analyzing data to build potential ML models, data scientists typically explore a variety of correlations that they hope may add a relevant piece to solving a given problem. In doing so, data scientists search for and evaluate potential (parts of) solutions that they aim to eventually capture in an ML model. For instance, data scientists of an online retailer might recognize drops in sales during world cup soccer matches and seek to incorporate according information into their ML model for sales prediction. Since these newly discovered correlations are not yet proven to represent causation, however, they must be thoroughly evaluated by domain experts attempting to integrate them with their existing solution descriptions. While this evaluation is often used for improving the input fed into the ML system, it can also yield novel explicit knowledge in the organization when domain experts deem the newly-discovered patterns to be promising (parts of) solutions and seek to combine them with their existing explicit knowledge to revise their standard practices. In particular, when data scientists experiment with ML approaches that may not be suitable for solving the problem (e.g., due to being too complex or computationally intensive), they may be able to draw previously-unnoticed connections from the iterative experimentation with the uncovered patterns (e.g., uncovering preferences of certain customer groups). Our experts emphasize that by reflecting on the adequacy of proposed patterns, domain experts often identify complementarities or discrepancies between their own rationales and the proposed solution patterns, allowing them to extend or detail their own solution approaches with additional aspects (e.g., adding new guidelines for handling exceptions or simplifications of their previous approach) or to identify and revise some of their potentially outdated or incorrect rationales:

> "*For the marketing team, it was also interesting because they knew, for example, that their marketing campaign had an impact on sales, but they didn't know how much of an impact. They were able to form an expectation that 30% off on shoes has more impact than 10% on hats. But they didn't know for sure that this was really the case and only anticipated it as one is 10%, the other is 30%, and they guessed that people buy more shoes than hats. But we had the data to try to correlate this.*" (I21)

> "*Just because I've seen the three classes in the past doesn't necessarily mean that a fourth one won't come along. [...] So if there's a fourth class that you can clearly see, hopefully you'll notice that. And if there's a class that you haven't even had on your radar yet, you might notice that because something is classified differently than you think. So I think humans have to continue to keep that in mind if that's relevant to their problem.*" (I3)

In this way, ML developments can help organizations evaluate, revise, and extend explicit knowledge by combining existing explicit knowledge with explicated insights into alternative solution approaches.

Internalization: Our experts note that the problem solving phase can also encourage domain experts to adopt solution patterns proposed by the data scientists and experiment with them in their respective domain. For instance, when data scientists, while working on an ML system to predict machine breakdowns, notice that a specific combination of machine parameters significantly heightens the chance of a failure, the engineers (i.e., domain experts) may want to run suitable tests on the machine to better understand the cause of the breakdowns and the conditions of when they may appear. Such experimentation can guide domain experts towards forming new tacit knowledge they can use to refine and extend their own solution practices when handling future problem instances. Thereby, expanding on the insights gained through internalization in the problem finding phase, further domain knowledge may be unearthed through the continued occupation with the problem and corresponding solutions:

> "*So in the beginning one store is compared with itself. Then, we can look how a particular item behaves in different stores, and then we can check how the whole market of this particular item behaves. And in each of these stages we use an algorithm to highlight what is not inline, and that is then manually reviewed again and, if necessary, either changed or sent back completely.*" (I16)

By exposing domain experts to potential data patterns, data scientists can also encourage them to think about the solution from a more pattern-driven perspective. Our experts report that domain experts often begin experimenting with the patterns to understand whether they can serve as (parts of) problem solutions, helping them to enrich their existing reasoning with unorthodox solution perspectives. For example, uncovered patterns can help to think about what other conditions might be considered to trigger a particular solution approach:

> "*Tinder is a good example. If you do not swipe—can you trap that as a signal? If you didn't like him or her, how can you take that and feed it back to the system?*" (I8)

In this way, ML developments can help organizations create, revise, and extend tacit knowledge by initiating and guiding experimentation with externalized data patterns in new directions.

Socialization: Our interviews reveal that in the process of finding appropriate solutions, data scientists often convene involved domain experts, ideally from different departments/areas, in knowledge sharing sessions, primarily to have domain experts collectively evaluate solution candidates proposed by data scientists. These sessions reflect opportunities for domain experts to meet with domain experts from the same or different areas to discuss how to appropriately solve a common problem. This allows domain experts to learn from each other through emergent discussions in which they share their perspectives on proposed solutions as well as their own best practices, fostering a culture free of silo thinking. For example, if data scientists are developing an ML system to provide sales reps with real-time advice about customers and are sharing emergent patterns that help close sales, the domain experts can share their own experiences of what works and what does not for them from different perspectives in the emergent discussion (e.g., sales reps and product engineers share their impressions of solutions). Our experts emphasize that these sessions can spark interactions among domain experts that can broaden each expert's perspective by giving them insight into each other's solution practices, help resolve conflicting approaches, and combine complementary approaches to gradually build consensus:

> *"In the discussion about what these results mean and what we can do differently with them—'how can we rearrange our lines? How can we adapt our shift schedules? How can we better deploy our personnel?'—so the knowledge gain takes place in the discourse about the project results, so with the interpretation of the results or the intermediate results. As I said, we follow a phase model with different stop or go decisions along different iteration cycles. The discourse about the intermediate results along this development process is what represents the essential benefit for [the organization]—besides the fact that the final ML model improves something by X percentage points every day in operation afterwards."* (I42)

In this way, ML developments can help organizations share tacit knowledge about solutions across departments and areas of expertise, helping to resolve potential conflicts and make complementary knowledge more accessible.

Solution implementation

Once adequate ML systems are built, they are implemented into existing processes for productive use. With the emerging interactions between domain experts and developed ML systems when both operate in the same domain, our experts emphasized that ML systems act as the main stimulus for the learning processes of domain experts that occur in the solution implementation phase, enabled by the frequent confrontation of domain experts with the solution approaches generated and applied by the ML systems.

Externalization: Our experts note that the developed ML systems can be used to continuously capture the tacit knowledge of domain experts: By tracking domain experts' behavior in data (e.g., by recording their past decisions), ML systems can be enabled to continuously observe domain experts' behavior, which allows the developed ML systems to mimic domain experts' solution approaches and thereby approximate their applied tacit knowledge with patterns captured in ML models. Our experts underscore that this allows them to store in their ML systems knowledge that was previously held only in the hard-to-access mental models of individuals. Our experts further emphasize that when organizations use transparent ML models (i.e., by using inherently transparent ML models (e.g., decision trees) or Explainable AI (XAI) methods) in their ML development, the use of the resulting ML systems allows them to externalize and disseminate the captured tacit knowledge of domain experts and thereby make it usable by other experts (e.g., who can now adopt other experts' decision rules that are explicitly described in a decision tree). As a result, our experts underline that ML systems can facilitate the dissemination of existing domain knowledge to other members of the organization and protect this knowledge from loss (e.g., due to personnel turnover):

> "*In the past, employees had a great deal of knowledge in their heads, which they then somehow managed via Excel lists and knew that they now had to calculate up/down again or something, that would flow into the models. And then, of course, it is no longer so dependent on the employees—that sounds bad now and probably also fuels the fears of one or the other employee—but it is definitely no longer so dependent on the employees.*" (I10)

In this way, ML developments can help organizations improve their continuous capture and documentation of domain experts' tacit knowledge by approximating tacit knowledge with transparent ML systems.

Combination: According to our experts, domain experts often learn by comparing and integrating their solution approaches with the ML systems' approach and their provided explanations in their daily tasks (e.g., when faced with a problem instance, they may enrich their solution with the approach outlined and applied by a transparent ML system, such as an ML system that outlines effects of marketing campaign parameters it has observed and considered to derive its sales predictions). Yet, our experts stress that the intensity of this learning process depends on the relevance of the knowledge generated and the level of automation of the given use case. For example, while domain experts may see little value in learning about the rationale behind autonomously tagged images from an ML system, an ML system that predicts crash data for automotive product development may provide highly interesting information on how to stimulate secure new product design ideas. Our interviews show that when domain experts learn from the ML system in this way, they are trying to reconcile the new explicit knowledge created by the ML system with the explicit knowledge (e.g., descriptions of effective product designs) that already exists in their organization:

> "*When the model makes a decision that the administrator can't understand, there's a lot of skepticism at first. [...] And it is very, very important to discuss this and then, I would say, in 98% of the cases, you generally come to a common denominator where the specialist department also says: 'Yes, okay, you can see it that way, that makes sense.*'" (I9)

However, our experts caution that domain experts must remain critical in their evaluation of such confrontations by ML systems. Since ML systems can make erroneous predictions, domain experts must mitigate the risk of spreading erroneous ML-generated knowledge throughout the organization, which could otherwise replace correct human knowledge. Our experts therefore highlight the need for domain experts to continuously verify that ML systems' proposed inputs are indeed based on causality:

> "*Anyone who calls himself an expert must, of course, be capable of criticism and question these findings. [...] If I have a correlation, and I see it often enough... the more often you see it, the more you can at least believe that it's stable. But that still doesn't say anything about causality, because you should still think about common causes and things like that.*" (I3)

In this way, ML developments can help organizations evaluate, revise, and extend explicit knowledge by combining existing explicit domain knowledge with carefully evaluated insights from ML systems.

Internalization: Our experts emphasize that transparent ML systems (e.g., decision trees or applied XAI approaches) can provide domain experts with explicit rationales for how the ML systems would argue for performing certain routines, which can then trigger domain expert experimentation with these rationales. For example, when using ML systems to predict the effectiveness of security measures in a new product design, the ML system can supplement its estimated effectiveness of a given design with specific reasons for how it reached its decision. When the rationales differ from the domain experts' own solution approaches, our experts find that the domain experts do not just naively adopt the rationales as new approaches, but typically begin to experiment with the rationales, testing their actual performance in different contexts (e.g., trying rules that the ML system would use in certain scenarios instead of their own approaches). In this way, the domain experts can gain new experiences guided by the ML systems' approaches, which enables them to revise and extend their own solution approaches:

> *"This happens very often. Especially through the description of causal relationships [through XAI approaches], anomalies or peculiarities are revealed that you had not considered before. This is very, very helpful and is really my be-all and end-all: If you can learn something new from it, then this can already bring you a lot further, because you can try this knowledge as an engineer in future cases and projects."* (I25)

Yet, our experts caution that this process can also become vicious if the ML systems' erroneous reasoning creeps slowly and unnoticed into the domain experts' solutions as they interact with the ML systems and develop confidence in the veracity of the ML systems' described rationales through repeated interactions rather than through reasoned justification based on comparison with their own domain knowledge:

> "*The interaction is actually always a problem for me, because it can happen that people suddenly rest on it and say: 'Yes, that thing is always right anyway, so I don't do my job anymore.' There's the situation where you kind of say: 'Okay, I just trust that thing and I always do what it says.' [...] So it also works the other way around, that the model influences the human being. That also works. And to find out this interaction is very challenging. This is very, very problematic.*" (I19)

In this way, organizations can acquire new tacit knowledge as human experts (un)intentionally collect new experiences guided by the rationales offered by transparent ML systems.

Socialization: Our experts underscore the importance of domain experts also learning about the tacit knowledge stored in non-transparent ML systems by observing the ML systems' behavior. By observing how ML systems make decisions, domain experts can, over time, approximate to some extent the rationale used by the ML systems, thereby creating their own tacit knowledge from the experienced task performances by the ML systems, and thus gradually become socialized into the ML systems' problem-solving behavior. For example, doctors who use a non-transparent ML system that suggests diagnoses for a while may learn to anticipate when and how the ML system is likely to reach certain conclusions and incorporate these heuristics into their own decision-making rationale. Our experts believe that this process is important, not only because it can help ensure the quality of the ML systems being used (as domain experts typically also judge the correctness of the approximated rationale of the ML system), but also because it requires domain experts to continually reflect on and question the correctness and currency of their own rationales and to revise them when necessary. Once domain experts identify a mismatch between an ML system's rationale and their own, our experts emphasize that domain experts then often engage in a collective sensemaking process in which domain experts reach out to other domain experts and data scientists to understand why there are conflicts between the ML systems' rationale and their own and how to resolve them, with the goal of reaching a conclusion about which of the two rationales is better suited as an adequate solution or how to integrate both rationales to create a new superior solution, which typically involves intensive discussions between data scientists and domain experts and possible reconfigurations of affected ML systems:

> *"The old veterans sit there, so to speak, who know exactly what is important—or think they know what is important—and they have to say: 'Yes, it makes sense if the model says that if I now allow ten millimeters more forward displacement, it will have a positive effect on my bus acceleration'. So in the end, SHAP and LIME [i.e., XAI approaches] are used to discuss whether this prediction makes sense or not, whether the model itself makes sense or not. So that is then discussed. Together, locally."* (I25)

> "*We do that [i.e., observing ML systems' behavior], but we need time to do it, because to find these patterns is not so easy. You have to use these models at least one or two years in production to know some patterns, so that you can say: 'Ok, this model works like this and we found this pattern.'*" (I14)

> *"Before we fully changed the approach to a machine leaning approach, there were two or three months where the [domain experts] would still predict the sales and the machine learning system would also predict the sales—and then there was a comparison. So, they would check, because they knew what they*

would come up with and they would maybe see if they are not right or that the model is not working well for this kind of campaign that we have and we would discuss it with them." (I21)

In this way, ML developments can help organizations share and resolve complementary and conflicting tacit knowledge between their human experts and ML systems.

In sum, our experts reported a variety of learning processes stimulated by the confrontation of domain experts with data scientists and ML systems. These processes can provide several opportunities to contribute to the revision of existing and the creation of new explicit and tacit domain knowledge in the organization. With the need for domain experts to reflect on their own understanding of problems and adequate solutions, and to engage in mutual learning with others with whom they may not typically interact, the three phases of ML developments seem to provide powerful stimuli for the four key organizational learning processes. In particular, by confronting domain experts with the need to explain their understanding to non-experts (i.e., data scientists), to express and evaluate it in terms of clear data variables and patterns, and to resolve conflicts and experiment with the understandings and observations of others (e.g., other domain experts, data scientists, and even ML systems), ML developments can provide unique encounters for stimulating organizational learning.

Discussion

Organizational learning is a crucial process that lies at the heart of organizational behavior and is known to fundamentally control an organization's performance (e.g., Argote et al., 2021; March, 1991). As organizations should therefore be careful to optimize their organizational learning processes, decades of research have analyzed how organizations can increase their organizational learning effectiveness (e.g., Argote & Miron-Spektor, 2011; Huber, 1991). More recently, research has recognized that ML systems can participate as a new type of organizational learner alongside humans, which can strongly influence organizational learning (e.g., Argote et al., 2021; Sturm et al., 2021a). So far, a handful of studies have explored how productive ML systems can contribute knowledge to organizational learning through learning dynamics that may emerge between humans and ML systems (i.e., Balasubramanian et al., 2022; Lyytinen et al., 2021; Ransbotham et al., 2020; Seidel et al., 2019; Sturm et al., 2021a). With our study, we aimed to explore not only how this human-AI interplay can change organizational learning, but how the preceding and continuous ML developments can stimulate different valuable types of learning processes among the human members of organizations, even when ML systems are not yet being used productively.

To this end, our study offers several theoretical contributions. First, we add to the emerging discussion on the impact of ML on organizational learning (e.g., Argote et al., 2021; Seidel et al., 2019). While existing work primarily focuses on the influence of productively deployed ML systems (e.g., Balasubramanian et al., 2022; Sturm et al., 2021a), our study shows that understanding the impact of ML on organizational learning can benefit from a more holistic perspective that considers not only the direct impact of ML systems by observing their mutual influence with human learners, but also takes into account the interactions between humans that are required to realize ML systems in the first place. In this sense, our study provides initial empirical evidence on how human-centered organizational learning processes unfold within ML developments, which we hope can provide a theoretical foundation for further research.

Second, our study demonstrates that ML developments can, indeed, serve as an important mechanism for stimulating organizational learning. The sub-processes uncovered by our study show how ML development activities, even if they do not result in productive ML systems, can add value by initiating different learning processes for domain experts. Our interviews highlight that ML developments can thus act as a new kind of "field of interaction" (i.e., as proposed by Nonaka (1994) as an important learning mechanism driven by encounters) where domain experts, lay people (i.e., data scientists), and ML systems can meet to collectively share, reflect, and revise organizations' domain knowledge. Since the stimulated learning processes do not only reside in ML developments, but domain experts are likely to disseminate their new knowledge to other parts of the organization when they return to their domain-specific departments (e.g., traders involved in developing a trading ML system applying their newly acquired knowledge in their trading activities and sharing it with other traders), our study suggests that the consequences of the identified learning processes may have a large impact on organizations' knowledge stock in the long run. However, while our study confirms the existence of these processes, their actual consequences for organizational learning remain unknown and deserve further attention in future studies. Moreover, uncovering the conditions that benefit or harm the identified learning processes may also provide much-needed insights into how to effectively coordinate ML developments and their consequences for improving organizational learning. To this end, while our experts emphasized the positive side of ML developments as learning stimuli, it may be helpful to further understand if and how they can also turn vicious to (other) organizational learning processes, and act as an alternative, obstacle, or complement to other stimulating mechanisms (e.g., rotational programs, interorganizational partnerships; e.g., Argote et al., 2021; Nonaka, 1994).

Third, the uncovered processes highlight that ML developments involve two different types of stimuli depending on the ML development phase: the data scientists and the developed ML systems. By confronting domain experts with new insights, emerging ambiguities, needed explanations, and their generated hypotheses and models, the interaction of domain experts with data scientists or ML systems provides opportunities for domain experts to substantially reflect on, revise, and articulate their knowledge. Our interviews show that these processes can help improve the availability and quality of organizations' domain knowledge (e.g., by forcing the articulation of tacit knowledge, triggering the revision of outdated knowledge). This finding reflects a valuable observation, because a well-known major problem in organizational learning research is the scarcity of activities that enable reflection and substantial revision of conventional knowledge (e.g., the tendency to favor exploitation of known over exploration of unknown solutions (Levinthal & March, 1993) and lack of questioning the adequacy of established learning conditions and goals through "double-loop learning" (Argyris, 1976)). ML developments may thus serve as a useful new mechanism that organizations can deliberately use to promote such reflective and revisionist activities, thereby helping to liberate organizations from suboptimal or outdated conventional knowledge (see, e.g., Argote et al., 2021; Argyris, 1976; Levinthal & March, 1993). While our study uncovers ML developments as a fruitful mechanism, research may greatly benefit from further studies on how organizations can effectively coordinate the interaction between domain experts, data scientists, and ML systems to integrate ML developments as a strategic element that fosters organization-wide domain knowledge.

Fourth, our study shows how ML systems can affect different types of knowledge (i.e., explicit and tacit knowledge). In particular, the role of ML systems as carriers of tacit knowledge has recently received increasing attention in research, stressing the potential role of ML systems as tacit knowledge repositories that can help prevent certain knowledge losses (e.g., Hadjimichael & Tsoukas, 2019; Lebovitz et al., 2021). In line with this research, our study confirms that ML systems are used as tacit knowledge repositories across different domains and industries. Interestingly, our study shows that ML systems can further serve as a powerful tool for externalizing tacit knowledge (e.g., a decision tree that mimics and describes experts' approaches) and as a stimulus for human learning of tacit knowledge. ML developments can thus provide an important mechanism to facilitate the management of tacit knowledge, which is known to be a difficult and tedious endeavor (e.g., Argote et al., 2021; Nonaka, 1994). Here, more research is needed to understand how organizations can effectively use ML systems to convert between tacit and explicit knowledge while avoiding losses and biases in the translation between humans' and/or ML systems' knowledge.

From a practical perspective, our study emphasizes that organizations should be cautious about reducing investment in ML development, even if they repeatedly fail to develop ML systems for productive use. Since most "unsuccessful" ML developments still involve the first two phases of ML development (i.e., problem finding and solving), organizations may otherwise miss out on significant benefits that can result from the learning processes involved—and thus miss a powerful driver that can improve their long-term knowledge needed to differentiate themselves from their competitors. In addition, our interviews emphasize that organizations should ensure that domain experts are well integrated into ML developments. Only when domain experts have the opportunity to interact with and be confronted by the insights of data scientists and ML systems can domain experts bring new insights back to their domains and infuse the new knowledge into further organizational learning processes, thereby spreading gained knowledge throughout the organization. Finally, organizations should be keen to allocate additional time and resources to ML developments to allow space for domain experts' analysis of insights and discussions that may not contribute to the development of the planned ML systems, but are focused on exploring potential process failures and resolving identified conflicts and ambiguities in existing knowledge. Otherwise, organizations may stifle valuable learning processes that may ultimately prove to be more than just a nice-to-have byproduct, but an essential stimulus for vital performance gains in the long run.

Of course, our study has several limitations. First, although our respondents cover a wide range of experiences across industries, roles, and ML use cases, there may still be data biases that we could not completely eliminate. Further qualitative and quantitative studies in different contexts may therefore help to uncover such biases and validate the applicability of the identified processes. Second, because we based our analyses exclusively on participants' post-hoc descriptions, our findings can only consider the details that were remembered and deemed relevant by our participants. In particular, in-depth case studies of ML developments could help to provide complementary insights into the learning processes uncovered in our study that we were unable to observe in the interviews. Finally, while we attempted to capture experiences with different types of ML systems across different industries, domains, and roles of participants, the processes identified may vary depending on the ML system under development. Here, future research could help validate and contextualize the learning processes for ML developments of specific types of ML systems.

Our study is only a first step in understanding the impact of ML developments on the crucial processes of organizational learning. As more research is needed to better understand how ML developments affect organizational learning and how ML developments can be used as a strategic means to improve organizational performance in the long run, we hope that our study itself can serve as a fruitful theoretical "stimulus" for future research to help rethink organizational learning theory in the era of AI.

Acknowledgements: This research was funded by the German Federal Ministry of Education and Research (BMBF) within the "Innovations for Tomorrow's Production, Services, and Work" program (funding number 02L19C150) coordinated by the Project Management Agency Karlsruhe (PTKA).

References

Alavi, M., & Leidner, D. E. (2001). Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Quarterly*, *25*(1), 107–136.

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., & Zimmermann, T. (2019). Software engineering for machine learning: A case study. *Proceedings of the 41st IEEE/ACM International Conference on Software Engineering*, 291–300.

Argote, L., Lee, S., & Park, J. (2021). Organizational learning processes and outcomes: Major findings and future research directions. *Management Science*, *67*(9), 5399–5429.

Argote, L., & Miron-Spektor, E. (2011). Organizational learning: From experience to knowledge. *Organization Science*, *22*(5), 1123–1137.

Argyris, C. (1976). Single-loop and double-loop models in research on decision making. *Administrative Science Quarterly*, *21*(3), 363–375.

Azevedo, A., & Santos, M. F. (2008). KDD, SEMMA and CRISP-DM: A parallel overview. *Proceedings of the IADIS European Conference on Data Mining*, 182–185.

Balasubramanian, N., Ye, Y., & Xu, M. (2022). Substituting human decision-making with machine learning: Implications for organizational learning. *Academy of Management Review*, *47*(3), 448–465.

Basadur, M., Graen, G. B., & Green, S. G. (1982). Training in creative problem solving: Effects on ideation and problem finding and solving in an industrial research organization. *Organizational Behavior and Human Performance*, *30*, 41–70.

Benbya, H., Pachidi, S., & Jarvenpaa, S. L. (2021). Special issue editorial: Artificial intelligence in organizations: Implications for information systems research. *Journal of the Association for Information Systems*, *22*(2), 281–303.

Berente, N., Gu, B., Recker, J., & Santhanam, R. (2021). Special issue editor's comments: Managing artificial intelligence. *MIS Quarterly*, *45*(3), 1433–1450.

Brynjolfsson, E., & Mitchell, T. (2017). What can machine learning do? Workforce implications. *Science*, *358*(6370), 1530–1534.

Butts, M. M., Becker, W. J., & Boswell, W. R. (2015). Hot buttons and time sinks: The effects of electronic communication during nonwork time on emotions and work-nonwork conflict. *Academy of Management Journal*, *58*(3), 763–788.

Davison, R. M., & Martinsons, M. G. (2016). Context is king! Considering particularism in research design and reporting. *Journal of Information Technology*, *31*(3), 241–249.

Fang, C., Lee, J., & Schilling, M. (2010). Balancing exploration and exploitation through structural design: The isolation of subgroups and organizational learning. *Organization Science*, *21*(3), 625–642.

Fiol, C., & Lyles, M. (1985). Organizational learning. *Academy of Management Review*, *10*(4), 803–813.

Flick, U. (2004). Triangulation in qualitative research. In U. Flick, E. von Kardoff, & I. Steinke (Eds.), *A companion to qualitative research* (pp. 178–183). Sage.

Hadjimichael, D., & Tsoukas, H. (2019). Toward a better understanding of tacit knowledge in organizations: Taking stock and moving forward. *Academy of Management Annals*, *13*(2), 672–703.
Hsieh, H.-F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research*, *15*(9), 1277–1288.
Huber, G. P. (1991). Organizational learning: The contributing processes and the literatures. *Organization Science*, *2*(1), 88–115.
Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255–260.
Kane, G. C., & Alavi, M. (2007). Information technology and organizational learning: An investigation of exploration and exploitation processes. *Organization Science*, *18*(5), 796–812.
Lebovitz, S., Levina, N., & Lifshitz-Assaf, H. (2021). Is AI ground truth really true? The dangers of training and evaluating AI tools based on experts' know-what. *MIS Quarterly*, *45*(3b), 1501–1525.
Levinthal, D. A., & March, J. G. (1993). The myopia of learning. *Strategic Management Journal*, *14*(S2), 95–112.
Levitt, B., & March, J. G. (1988). Organizational learning. *Annual Review of Sociology*, *14*, 319–338.
Lindebaum, D., Vesa, M., & den Hond, F. (2020). Insights from "the machine stops" to better understand rational assumptions in algorithmic decision making and its implications for organizations. *Academy of Management Review*, *45*(1), 247–263.
Lyytinen, K., Nickerson, J. V., & King, J. L. (2021). Metahuman systems = humans + machines that learn. *Journal of Information Technology*, *36*(4), 427–445.
March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization Science*, *2*(1), 71–87.
March, J. G. (2010). *The Ambiguities of experience*. Cornell University Press.
March, J. G. (2006). Rationality, foolishness, and adaptive intelligence. *Strategic Management Journal*, *27*(3), 201–214.
Martinez, I., Viles, E., & G. Olaizola, I. (2021). Data science methodologies: Current challenges and future approaches. *Big Data Research*, *24*.
Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez-Orallo, J., Kull, M., Lachiche, N., Ramirez-Quintana, M. J., & Flach, P. (2021). CRISP-DM Twenty years later: From data mining processes to data science trajectories. *IEEE Transactions on Knowledge and Data Engineering*, *33*(8), 3048–3061.
Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.
Myers, M. D., & Newman, M. (2007). The qualitative interview in IS research: Examining the craft. *Information and Organization*, *17*(1), 2–26.
Nonaka, I. (1994). A dynamic theory of organizational knowledge creation. *Organization Science*, *5*(1), 14–37.
Ransbotham, S., Khodabandeh, S., Kiron, D., Candelon, F., Chu, M., & LaFountain, B. (2020). Expanding AI's impact with organizational learning. *MIT Sloan Management Review, 62270*, 1–15.
Russell, S. J., & Norvig, P. (2021). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
Saldaña, J. (2015). *The coding manual for qualitative researchers* (3rd ed.). Sage.
Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. *Advances in Neural Information Processing Systems*, *28*.
Seidel, S., Berente, N., Lindberg, A., Lyytinen, K., & Nickerson, J. (2019). Autonomous tools and design: A triple-loop approach to human-machine learning. *Communications of the ACM*, *62*(1), 50–57.
Simon, H. (1991). Bounded rationality and organizational learning. *Organization Science*, *2*(1), 125–134.
Studer, S., Bui, T. B., Drescher, C., Hanuschkin, A., Winkler, L., Peters, S., & Müller, K. R. (2021). Towards CRISP-ML(Q): A machine learning process model with quality assurance methodology. *Machine Learning and Knowledge Extraction*, *3*(2), 392–413.
Sturm, T., Gerlach, J. P., Pumplun, L., Mesbah, N., Peters, F., Tauchert, C., Nan, N., & Buxmann, P. (2021a). Coordinating human and machine learning for effective organizational learning. *MIS Quarterly*, *45*(3), 1581–1602.
Sturm, T., Koppe, T., Scholz, Y., & Buxmann, P. (2021b). The case of human-machine trading as bilateral organizational learning. *Proceedings of the 42nd International Conference on Information Systems*.
Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, *1*, 29–39.