Data Sensitivity and Domain Specificity in Reuse of Machine Learning Applications

Corinna Rutschi[1] · Nicholas Berente[2] · Frederick Nwanganga[2]

Abstract

Data sensitivity and domain specificity challenges arise in reuse of machine learning applications. We identify four types of machine learning applications based on different reuse strategies: generic, distinctive, selective, and exclusive. We conclude with lessons for developing and deploying machine learning applications.

Keywords Machine learning · Data sensitivity · Domain specificity · Reuse

1 Introduction

In this paper, we introduce a framework that can help think about important reuse challenges when developing and training machine learning technologies. We distinguish machine learning applications based on the *sensitivity of the data* used to train the underlying machine learning model and the *model's specificity* to a particular organizational domain. We draw on examples from chatbot machine learning applications to illustrate.

Organizations often reuse machine learning applications (or components of them) across contexts. Reuse is said to accelerate training and strengthen machine learning models (Alon et al., 2019; Ning, Guan, & Shen, 2019). Reuse is said to help mitigate the high costs of acquiring reliable training data (Denning & Denning, 2020). But reuse is not simple (McCarthy & Hayes, 1969).

Reuse introduces a variety of tradeoffs with privacy, openness, and customization, and with different management challenges, including the free-rider problem (Gross & De Dreu, 2019; Holmstrom, 1982; Panchanathan & Boyd, 2004; Sanghavi & Hajek, 2008), the frame problem (Dennett, 1984; McCarthy & Hayes, 1969; Salovaara, Lyytinen, & Penttinen, 2019), privacy (Abadi et al., 2016; Cheng, Liu, Chen, & Yang, 2020; Dwork & Roth, 2014; Osia, Shamsabadi, Taheri, Rabiee, & Haddadi, 2018; Yang, Liu, Chen, & Tong, 2019), and building capabilities (Berente et al., 2021). Natural language processing (NLP) chatbot applications[1] are the example that we draw on for understanding challenges to reuse more generally. A chatbot is a program used to communicate with humans through natural language, via voice or text input (Abdul-Kader & Woods, 2015; Patil, Marimuthu, & Niranchana, 2017; Weizenbaum, 1966).

Chatbots typically involve NLP machine learning algorithms for continuous machine learning based on data that refine the underlying model (i.e., the dialog structures) and improve how chatbots handle conversations and tasks (Baird & Maruping, 2021; Quamar et al., 2020). Chatbots handle simple, frequently asked questions (FAQs) or ambitious customer service and sales issues. It can provide critical aid to digital assistants (the most famous being Amazon's Alexa and Apple's Siri; Abdul-Kader & Woods, 2015; Carlander-Reuterfelt et al., 2015; Patil et al., 2017; Willcocks & Lacity, 2016). However, different chatbots require different development and training approaches (Gao et al., 2018). Two key dimensions shape unique reuse requirements for machine

✉ Corinna Rutschi
  corinna.rutschi@unibe.ch

  Nicholas Berente
  nberente@nd.edu

  Frederick Nwanganga
  fnwangan@nd.edu

1 University of Bern, Bern, Switzerland

2 University of Notre Dame, Mendoza College of Business
   Notre Dame, Indiana, USA

---

[1] We base our examples on a series of research projects with chatbot software development organizations, and their implementation of these technologies across a variety of organizational domains, predominantly in the European financial industry.

Table 1 Data Sensitivity and Domain Specificity of Machine Learning Applications

Key Concerns in Reuse of Machine Learning Applications across Domains	
Data Sensitivity	Who can and should have access to data? What privacy regulations are needed? Is there competitive advantage in attending to this?
Domain Specificity	How reusable and generalizable? Will it work in other domains? How similar do the domains have to be?

learning applications: data sensitivity and domain specificity (see Table 1).[2]

By data sensitivity we mean the degree to which data needs to be protected by judgment or regulation (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Meurisch & Mühlhäuser, 2021; Osia et al., 2018; Yang et al., 2019). Data sensitivity in machine learning is the degree to which training data should be protected and remain private. For chatbots, training data usually refers to conversation content organizations use to train the underlying chatbot dialog model or algorithm. Sensitive data might involve personal or confidential organizational or customer information and be subject to strict legal data protection policies (e.g., the European General Data Protection Regulation, or GDPR).[3] Organizational data might be sensitive because, falling into the wrong hands, the impacted organization loses competitive advantage. Data are confidential if subject to regulation, are critical assets to an organization, or are something a stakeholder is uncomfortable sharing (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Meurisch & Mühlhäuser, 2021; Osia et al., 2018; Yang et al., 2019). Less sensitive data include publicly available information, or other information organizations are willing to share. Data are open if not critical to the organization, not private, protected, or important to competitive advantage.

By domain specificity we refer to the degree to which something's performance depends on context. If not context-specific to a domain, the application can be used in broader situations. Domain specificity determines whether organizations can meaningfully use applications across domains. Domain specificity can affect competitive advantage. In the chatbot example, a specific custom application that uses dialog structures configured and trained for a specific domain cannot easily be reused. On the other hand, general models that involve non-proprietary applications might be widely applicable, supplying inexpensive solutions for small talk conversation functionality in different domains.

Table 2 Four Types of Machine Learning Applications

		Data Sensitivity	
		Open Data (Public)	Confidential Data (Private)
Domain Specificity	General Model (General Purpose)	**Generic Applications** General model, open data *Example*: small talk *Reuse management challenge*: The Free-Rider Problem	**Selective Applications** General model, confidential data *Example*: help desk *Reuse management challenge*: Privacy and Regulations
	Custom Model (Specific Purpose)	**Distinctive Applications** Custom model, open data *Example*: FAQs *Reuse management challenge*: Frame Problem	**Exclusive Applications** Custom model, confidential data *Example*: PDA *Reuse management challenge*: ML capabilities

2 Different Types of Machine Learning

Drawing on this distinction between data sensitivity and domain specificity, we classify four types of machine learning applications: generic, distinctive, selective, and exclusive (see Table 2). Next we briefly introduce each.

2.1 Generic Machine Learning Applications

Generic applications involve general-purpose models trained on open or public data. For NLP technologies, this means multi-purpose, everyday conversations. These conversations can generalize across a variety of domains. The underlying dialog model is non-specific and open. Organizations can potentially reuse it as a predefined template for different domains. Thus, organizations can develop generic applications based on predefined models without starting from the ground up but reproducing such applications typically at lower costs across domains. Software developers can freely share generic applications across their client base, bringing costs down and rapidly improving functionality over time. Assuming that organizations are willing to share their open models and datasets across various contexts, the development and training of other similar generic applications can be accelerated.

One typical example of a generic chatbot application is the small talk functionality part of virtually every chatbot. Organizations can implement and reuse small talk functionality context-independently while it improves the overall user experience. For example, the user could ask the chatbot, "How are you?" whereupon the chatbot would answer reasonably (see Fig. 1). Once small talk functionality is developed for one organization, it can be readily applied to

---

[2] This is analogous, but not directly related to sensitivity and specificity as statistical measures used in the evaluation of prediction models.

[3] https://gdpr.eu/.

others. Thus, there is nothing particularly proprietary about either the functionality of generic machine learning applications (e.g., dialog models of chatbots) or the data (e.g., the content of the conversations a chatbot can conduct). The more organizations adopt the same generic application, the better it becomes, and the more reusable its model and data are across domains and organizations.

In investigating a chatbot project at a European bank, we observed how the bank introduced a customer-facing chatbot that could initially conduct small talk in German. Later they implemented a similar chatbot in French. When implementing the French chatbot, the project team built on existing generic small talk dialogs and transferred them from one language context (i.e., German) to the new (i.e., French).

The key management challenge for generic applications involves what is often described as the "free-rider" problem – organizations benefit from the sharing of others but do not share themselves (thus flouting norms of reciprocity). Free-riding in a variety of contexts is a classic behavioral problem, and there are numerous proposed solutions – including incentives, competition, punishment, and social pressure – but each approach has both advantages and disadvantages (Gross & De Dreu, 2019; Holmstrom, 1982; Panchanathan & Boyd, 2004; Sanghavi & Hajek, 2008). Development organizations need to develop tactics to deal with the free-rider problem and actively manage free-riding in reuse. Incentives or requirements for sharing as part of the contract for the use of the application are often the first steps, but this process needs to be actively monitored, coordinated, and updated as requirements change.
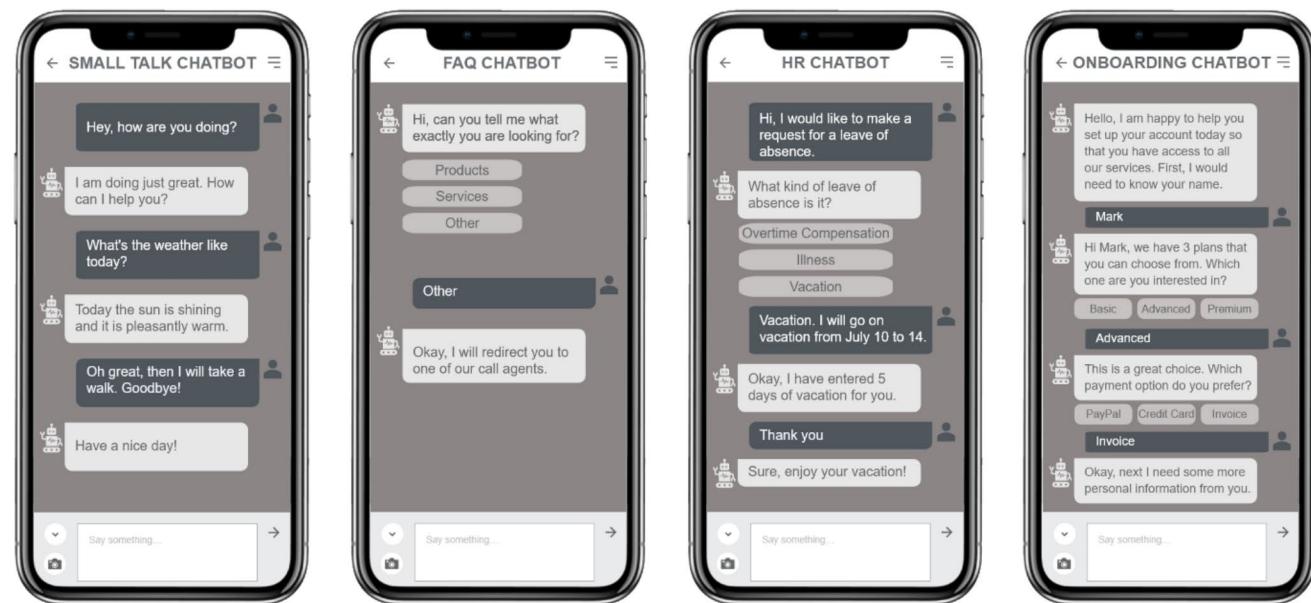
2.2 Distinctive Machine Learning Applications

Distinctive applications are customized and applicable to a particular context, but the data they are trained on is not sensitive. Thus, we characterize distinctive applications by a custom model and open data. However, in most cases, organizations cannot reuse the training data to develop or train other similar applications due to its specificity to a particular domain.

For chatbots, distinctive applications involve conversations where the underlying dialog model is highly specific but open since the data is freely available, non-confidential, or public. Although the dialog model is open and organizations could potentially reuse it, they often do not reuse it because it is not particularly useful in any other domain. Examples of distinctive chatbot applications include frequently asked questions (FAQ) or marketing bots. FAQ bots (see Fig. 1) are typically quite contextually dependent, even for organizations operating in similar domains. Frequently asked questions and corresponding answers are mostly freely accessible on an organization's website or intranet. However, even though the dialog model for FAQs is open, it is often highly customized according to the domain or organization.

An example of a distinctive application we came across was a marketing chatbot at an organization in the financial services industry. The organization wanted to introduce a chatbot for its employees to show them all current promotions that they could offer their customers. The employees engaging with customers on the telephone could ask the chatbot what promotions to offer to the respective customers. Since both the nature and content of such promotions

Fig. 1 Example Chatbot Dialogues

vary from domain to domain and from organization to organization, the dialog model was not reusable to develop and train other similar applications.

Developers must understand when particular machine learning applications do and do not hold for new domains. This issue of applicability beyond the training domain has been dubbed the "frame problem" and is perennially one of the strongest challenges to artificial intelligence technologies (Dennett, 1984; McCarthy & Hayes, 1969). Technical solutions to the frame problem do exist, usually associated with some form of transfer learning (Pan & Yang, 2009) or general, versatile model development (Hernández-Orallo et al., 2016), but organizations must implement these solutions mindfully and continually monitor and reappraise them, or they risk running into serious problems with accuracy, reliability, and security (Salovaara et al., 2019).

2.3 Selective Machine Learning Applications

Selective applications build on similar models across organizations but involve private and context-dependent training datasets. Thus, we characterize selective applications by an open model but confidential data.

In the case of chatbots, selective applications conduct more complex conversations that are similar for organizations operating in the same industries or domains. The underlying dialog model is widely applicable. Organizations may reuse it across similar domains. This may allow organizations to develop this type of application more quickly based on a predefined skeleton or template without being obliged to start from the ground up. In contrast, the data is highly sensitive and therefore private. As a result, organizations cannot or are unwilling to make their data available for the development and training of other similar applications in different domains.

We encountered a human resources (HR) chatbot application (see Fig. 1) that could respond to an organization's employees on common questions related to HR policies and procedures and complete specific service fulfillment tasks. For example, the chatbot could handle the fulfillment task of issuing HR letters for passport applications. The HR chatbot could create such letters by requesting all necessary information from the employee, entering it into a dedicated system to generate the required HR document, which the chatbot then sent back to the employee.

The same selective applications often can be meaningfully applied in multiple domains or organizations. Although organizations typically cannot reuse data of selective applications across domains, certain practices, such as data anonymization or encryption, allow some potential for model reuse (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Meurisch & Mühlhäuser, 2021; Osia et al.,

2018; Yang et al., 2019). Alternatively, approaches of centralized model training can be helpful, where the model is trained on the data at a central location without exposing sensitive data. Therefore, sub-models can be trained based on local data available in different departments of an organization or different organizations. These sub-models can then be assembled into a larger global model. Different departments of the same organization or different organizations can thereby train a model collectively using approaches such as federated learning and data minimization, among others (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Osia et al., 2018; Yang et al., 2019). Different approaches for handling confidential data can sometimes be combined (Meurisch & Mühlhäuser, 2021). However, before applying the technical solution to reuse training data, it is essential to understand its organizational and regulatory implications. Regulations and organizational policies are continually changing and require that machine learning organizations stay on top of such changes to adjust approaches accordingly (Berente et al., 2021).

2.4 Exclusive Machine Learning Applications

Exclusive applications build on private custom models trained on private data. These are proprietary, unique applications that conduct sophisticated conversations tailored to specific domains. The underlying model and the data are highly specific, sensitive, and thus private. Therefore, organizations can reuse neither the model nor the data. This type of application needs to be developed individually from the ground up for every domain or organization.

The machine learning model and data of exclusive applications are confidential and often different for different domains or organizations. Either an organization cannot meaningfully reuse models, or other organizations do not allow them reusing their models or data.

The chatbot vendor we engaged with implemented a customer onboarding chatbot (see Fig. 1) for one of its clients. The chatbot interacted with a potential customer through a dialog interface while gathering all required information and documents to onboard the particular customer. With the help of proprietary machine learning techniques such as computer vision, the chatbot could automatically verify documents such as a customer's passport, address, or credit score documents. The company saw its chatbot as an asset to improve the customer journey. Therefore, the chatbot provided them a competitive advantage, and they did not want to share the underlying model or the data.

The challenge with exclusive applications is that developing them requires more effort and a good deal of confidential data. The sample size of the training data for a single domain is often limited, and the resulting application

inherits those limitations. Without widespread datasets and general-purpose models to draw from, organizations need to build the capabilities internally to develop and deploy such exclusive applications or gain the involvement of, and dependence on, consulting partnerships with developers that have these capabilities. The management of machine learning model development and deployment is no trivial task. Organizations need to build capabilities to manage the issues of autonomy, inscrutability, and learning to continually evolve these capabilities to stay ahead of performance, security, reliability, and ethical issues (Berente et al., 2021).

3 Managing Multiple Machine Learning Applications

Thus, it is clear that different machine learning applications require different management approaches for managing reuse. These management issues increase when different kinds of machine learning applications are combined to form a more complex system (or machine learning solution) that incorporates more functionality.

For example, virtually every chatbot incorporates generic functionality such as small talk – even when organizations do not think they need this functionality at first, often chatbots evolve to include it over time. In one situation, we ran across another European bank that implemented an FAQ chatbot to handle questions that employees would frequently ask. At first, the bank did not think that users would be interested in small talk. However, it soon became apparent that the employees complained about an insufficient user experience. As soon as they introduced small talk functionality, the employees' use of the chatbot increased, and they reported positive experiences when interacting with the chatbot.

Thus, as one form of a machine learning system, chatbots can be composed of multiple applications. An example of such a chatbot could be an IT helpdesk chatbot (see Fig. 2), which handles requests related to IT issues such as password resets or unlocks. An IT helpdesk chatbot could consist of generic applications such as small talk functionality,

distinctive applications such as FAQ functionality (to respond to general IT-related questions), selective applications such as account management functionality (to respond, for example, to 'can you change my account settings?'), and exclusive applications such as access management functionality (to respond, for example, to 'can you reset my password?'). Assuming that another organization plans to implement a similar chatbot, they could potentially reuse the dialog models and the data of some applications (i.e., generic, distinctive, and selective) to speed up the development of the new chatbot.

Depending on the type of machine learning application (i.e., generic, distinctive, selective, and exclusive), organizations can partially reuse machine learning models and training data directly or with the help of specific adaptation approaches (e.g., through anonymization of confidential data). In some cases, however, models or data cannot be reused at all due to high levels of domain specificity and data sensitivity.

Overall, organizations need to manage reuse in different ways for different machine learning applications, depending on the particular requirements of the type of application; this involves dealing with free-rider problem (Gross & De Dreu, 2019; Holmstrom, 1982; Panchanathan & Boyd, 2004; Sanghavi & Hajek, 2008) in generic applications, the frame problem (Dennett, 1984; McCarthy & Hayes, 1969; Salovaara et al., 2019) in distinctive applications, privacy (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Osia et al., 2018; Yang et al., 2019) in selective applications, and building capabilities (Berente et al., 2021) in exclusive applications.

4 Lessons for Machine Learning Reuse

From our analysis, we derive that there are (at least) four general types of machine learning applications regarding data sensitivity and domain specificity. Each of these applications involves different management challenges. Generic applications require that organizations deal with the free-rider problem (Gross & De Dreu, 2019; Holmstrom, 1982; Panchanathan & Boyd, 2004; Sanghavi & Hajek, 2008), whereas distinctive applications involve navigating the frame problem (Dennett, 1984; McCarthy & Hayes, 1969; Salovaara et al., 2019). Selective applications typically require some technical means for dealing with private data (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Osia et al., 2018; Yang et al., 2019), and exclusive applications necessitate that organizations build the complete set of capabilities to develop and manage machine learning applications (Berente et al., 2021).
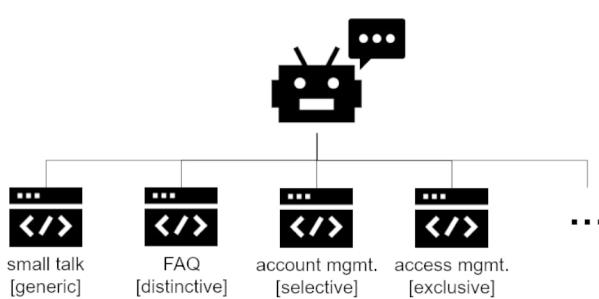

Fig. 2 IT Helpdesk Bot with Multiple Applications

As our IT help desk chatbot example illustrates, combining several or all four types of machine learning applications in the same system or machine learning solution is not uncommon. Reflecting on what we learned from studying various chatbot implementations, we suggest that organizations consider the following lessons when developing and deploying machine learning solutions: (1) categorize use cases in terms of domain specificity, (2) categorize use cases in terms of data sensitivity; (3) adopt a modular design philosophy; and (4) define strategies for reuse. We discuss each of the four lessons we propose in the following.

4.1 Lesson 1: Categorize Use Cases in Terms of Domain Specificity

Before organizations introduce machine learning solutions, they must define the domain in which they aim to deploy the underlying machine learning applications. Specifically, this requires a clear understanding of how generic or specific the domain of use is. Instead of thinking of machine learning solutions as one complex construct, they should be thought of in terms of specific use cases or applications. This enables those managing machine learning applications to map solutions and process automation to specific requirements intentionally. To be successful, organizations must conduct a thorough assessment of the environment (i.e., the domain) in which they want to automate, understand stakeholder expectations, and consider organizational policies to properly categorize the domain specificity of a particular machine learning application.

However, organizations must take care not to fall victim to the frame problem, which was introduced as an issue related explicitly to distinctive machine learning applications (Dennett, 1984; McCarthy & Hayes, 1969). In such a scenario, an application would become incoherent when queried on a subject matter because it would have difficulty distinguishing between relevant and unnecessary information in its source database for a particular domain. To overcome this challenge, the machine learning application would need a certain amount of information (or data) about a varied set of domains to determine what data is relevant to the context of each domain, and managers and developers would need specific knowledge about the boundaries of a model's applicability.

4.2 Lesson 2: Categorize Use Cases in Terms of Data Sensitivity

Machine learning applications generally benefit from training on large datasets. As a result, organizations will often tend to collect as large a dataset as possible, especially when building an application shared across domains. In such a

situation, organizations may want to benefit from other organizations' external data to validate their internally trained models. However, most organizations have the incentive to keep sensitive data private for obvious reasons (e.g., regulations or competitive advantages). Even non-sensitive data in the wrong hands can negatively impact an organization. This is a fundamental challenge with training machine learning applications using data shared across organizations. Privacy and regulatory concerns arise primarily in connection with selective machine learning applications (Abadi et al., 2016; Cheng et al., 2020; Dwork & Roth, 2014; Osia et al., 2018; Yang et al., 2019). Overcoming this challenge requires that organizations clearly categorize a machine learning solution's degree of data sensitivity before developing data governance and privacy policies around data use and sharing.

In this regard, some of the common approaches that an organization can adopt include removing personally identifiable data, anonymizing or encrypting data, randomizing responses, data minimization, and the use of differential privacy or federated learning (Cheng et al., 2020; Dwork & Roth, 2014; Meurisch & Mühlhäuser, 2021; Osia et al., 2018; Yang et al., 2019). These policies or approaches should not only be relegated to the implementation team. They are strategic decisions that should be made and adopted at an organizational level. However, organizations must also be careful regarding what data they (re)use to train their models. The same model trained on different data can perform vastly different and thus produce different outputs, which is not always beneficial and can sometimes be dangerous or compromise the model's explanatory power (Asatiani et al., 2021). Thus, organizations face the challenge of having sufficient and having the 'right' training data. The framework we provide in this paper can help organizations categorize machine learning use cases by domain specificity (Lesson 1) and data sensitivity (Lesson 2) as either generic, distinctive, selective, or exclusive.

4.3 Lesson 3: Adopt a Modular Design Philosophy

Once an organization successfully categorizes the use cases for the machine learning applications it intends to develop, it can develop modular components for each use case. Each component should be developed separately but built using a common technical architecture with standard interfaces for modules to communicate with each other (Sanchez & Mahoney, 1996; Simon, 1962). A particular machine learning solution can include multiple modules (i.e., applications), each of which can be classified differently in terms of domain specificity and data sensitivity. As a result, each module could be treated differently regarding how to use and train it and what data it accesses. This modular way of organizing machine learning applications enables the reuse

of specific components for future machine learning implementations. Moreover, such a modular approach can also be considered a capability to develop and deploy more complex machine learning solutions. Often organizations cannot directly reuse all components of a particular machine learning solution, mostly not exclusive machine learning applications, but they may reuse particular generic, distinctive, or selective applications.

4.4 Lesson 4: Define Strategies for Reuse

By developing and managing machine learning solutions in a modular way, organizations can pick and choose which previously developed components (or applications) to include in a new solution. Thereby, they can avoid wasted effort by not having to develop entire solutions from the ground up every time. This is especially salient for new solutions with significant overlap, in terms of the domain of use and training data, with other previously developed solutions. To facilitate reuse, organizations should establish version-controlled centralized repositories for checking in and checking out models and data. They should also develop policies and standards for when and how they can reuse existing components. Every new solution should be structured in such a way that these standards are used as a scaffold on which to leverage existing or yet-to-be-developed applications to build a more powerful machine learning solution. This scaffolding approach to solution development is powerful and well established in software development practice, especially when creating and reusing boilerplate code, and this approach is clearly applicable to the reuse and recombination of machine learning applications.

Leveraging reusable components across organizations is highly advantageous in developing and deploying machine learning applications. However, care must be taken to avoid or mitigate the impact of the free-rider problem, particularly with open source or collaborative projects. To mitigate this problem, efforts should be made to incentivize organizations to contribute models or data in order to foster reuse (Gross & De Dreu, 2019; Holmstrom, 1982; Panchanathan & Boyd, 2004; Sanghavi & Hajek, 2008). A possible policy framework for success are Ostrom's (Ostrom, 1990) design principles for common-pool resources. Such a framework would establish clear rules for who has access to models or data; how they are shared; how responsibility is shared for the creation, acquisition, and maintenance of models and data; what penalties are levied against rule-breakers; how rules are enforced; how rules are modified; and how conflicts are resolved.

5 Conclusion

Machine learning solutions have become widespread. Chatbots are often part of an organization's very first experiences with machine learning. Therefore, gleaning lessons from the implementation of chatbot applications can help organizations understand some key lessons that can apply to the host of machine learning applications. Machine learning techniques are becoming increasingly critical to various applications and will not be going away anytime soon. Organizations must start building a fundamental understanding of developing, training, and implementing different sorts of machine learning applications. Organizations must build up core capabilities around machine learning and prioritize machine learning models and training data governance. Those who experiment early on in implementing machine learning technologies will build the capabilities that will enable them to accelerate and enhance the overall development and training of machine learning applications over time (Berente et al., 2021).

6. References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). *Deep Learning with Differential Privacy* In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308–318.

Abdul-Kader, S. A., & Woods, J. C. (2015). Survey on Chatbot Design Techniques in Speech Conversation Systems. *International Journal of Advanced Computer Science and Applications*, 6(7), 72–80.

Alon, U., Zilberstein, M., Levy, O., & Yahav, E. (2019). *Code2vec: Learning Distributed Representations of Code* In Proceedings of the ACM on Programming Languages, 1–29.

Asatiani, A., Malo, P., Nagbøl, P. R., Penttinen, E., Rinta-Kahila, T., & Salovaara, A. (2021). Sociotechnical Envelopment of Artificial Intelligence: An Approach to Organizational Deployment of Inscrutable Artificial Intelligence Systems. *Journal of the Association for Information Systems*, 22(2), 325–352.

Baird, A., & Maruping, L. M. (2021). The Next Generation of Research on Is Use: A theoretical Framework of Delegation to and from Agentic is Artifacts. *MIS Quarterly*, 45(1), 315–341.

Berente, N., Gu, B., Recker, J., & Santhanam, R. (2021). Managing Artificial Intelligence. *MIS Quarterly*, *45*(3), 1433–1450.

Carlander-Reuterfelt, D., Carrera, Á., Iglesias, C. A., Araque, Ó., Rada, J. F. S., & Muñoz, S. (2015). Jaicob: A Data Science Chatbot. *Ieee Access : Practical Innovations, Open Solutions*, *8*, 180672–180680.

Cheng, Y., Liu, Y., Chen, T., & Yang, Q. (2020). Federated Learning for privacy-preserving Ai. *Communications of the ACM*, *63*(12), 33–36.

Dennett, D. C. (1984). Cognitive Wheels: The Frame Problem of Ai.Minds, machines and evolution,129–151.

Denning, P. J., & Denning, D. E. (2020). Dilemmas of Artificial Intelligence. *Communications of the ACM*, *63*(3), 22–24.

Dwork, C., & Roth, A. (2014). The algorithmic foundations of Differential privacy. *Theoretical Computer Science*, *9*(3–4), 211–407.

Gao, J., Galley, M., & Li, L. (2018). *Neural Approaches to Conversational Ai*. Paper presented at the Proceedings of ACM SIGIR conference (SIGIR 2018).

Gross, J., & De Dreu, C. K. (2019). Individual solutions to Shared problems create a modern tragedy of the Commons. *Science advances*, *5*(4), eaau7296.

Hernández-Orallo, J., Martínez-Usó, A., Prudêncio, R. B., Kull, M., Flach, P., Farhan Ahmed, C., & Lachiche, N. (2016). Reframing in Context: A systematic Approach for Model Reuse in Machine Learning. *AI Communications*, *29*(5), 551–566.

Holmstrom, B. (1982). Moral Hazard in Teams.The Bell Journal of Economics,324–340.

McCarthy, J., & Hayes, P. J. (1969). Some philosophical problems from the standpoint of Artificial Intelligence. In B. Meltzer, & D. Michie (Eds.), *Readings in Artificial Intelligence* (pp. 463–502). Edinburgh University Press.

Meurisch, C., & Mühlhäuser, M. (2021). Data Protection in Ai Services: A Survey. *ACM Computing Surveys (CSUR)*, *54*(2), 1–38.

Ning, L., Guan, H., & Shen, X. (2019). *Adaptive Deep Reuse: Accelerating Cnn Training on the Fly* In 2019 IEEE 35th International Conference on Data Engineering (ICDE), 1538–1549.

Osia, S. A., Shamsabadi, A. S., Taheri, A., Rabiee, H. R., & Haddadi, H. (2018). Private and scalable Personal Data Analytics using hybrid edge-to-Cloud Deep Learning. *Computer*, *51*(5), 42–49.

Ostrom, E. (1990). *Governing the Commons: The evolution of institutions for collective action*. Cambridge university press.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, *22*(10), 1345–1359.

Panchanathan, K., & Boyd, R. (2004). Indirect reciprocity can stabilize Cooperation without the Second-Order Free Rider Problem. *Nature*, *432*(7016), 499–502.

Patil, A., Marimuthu, K., & Niranchana, R. (2017). Comparative study of Cloud Platforms to develop a Chatbot. *International Journal of Engineering & Technology*, *6*(3), 57–61.

Quamar, A., Lei, C., Miller, D., Ozcan, F., Kreulen, J., Moore, R. J., & Efthymiou, V. (2020). *An Ontology-Based Conversation System for Knowledge Bases* In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, 361–376.

Salovaara, A., Lyytinen, K., & Penttinen, E. (2019). High reliability in Digital Organizing: Mindlessness, the Frame Problem, and Digital Operations. *MIS Quarterly*, *43*(2), 555–578.

Sanchez, R., & Mahoney, J. T. (1996). Modularity, flexibility, and Knowledge Management in product and Organization Design. *Strategic management journal*, *17*(S2), 63–76.

Sanghavi, S., & Hajek, B. (2008). A new mechanism for the Free-Rider Problem. *IEEE transactions on automatic control*, *53*(5), 1176–1183.

Simon, H. A. (1962). *The Architecture of Complexity* In Proceedings of the American Philosophical Society, 467–482.

Weizenbaum, J. (1966). Eliza—a computer program for the study of Natural Language Communication between Man and Machine. *Communications of the ACM*, *9*(1), 36–45.

Willcocks, L., & Lacity, M. C. (2016). *Service automation: Robots and the future of work*. Steve Brookes Publishing.

Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated Machine Learning: Concept and Applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *10*(2), 1–19.

Corinna Rutschi engages in strategic digital transformation and sustainability initiatives from an IT perspective at the Swiss engineering company Bühler Group. She received her Ph.D. in Information Systems from the University of Bern. Her research interests include digital agents (such as chatbots), artificial intelligence, software implementation, and interfaces.

Nicholas Berente is Professor of IT, Analytics, and Operations at the University of Notre Dame's Mendoza College of Business. He received his Ph.D. from Case Western Reserve University. His research interests include digital innovation, artificial intelligence, and institutional change in organizations. He is Senior Editor at *MIS Quarterly and at Information and Organization*.

Frederick Nwanganga is an Associate Teaching Professor of IT, Analytics, and Operations at the University of Notre Dame's Mendoza College of Business. He holds a Bachelor's degree as well as a Master's degree in Computer Science from Andrews University, a Master of Business Administration from Indiana University, and a Ph.D. in Computer Science and Engineering from the University of Notre Dame. Fred's primary areas of research and expertise include applied machine learning in enterprise IT service delivery, learning analytics, and data science education.