Proceedings of the Tenth AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2022)

A Human-Centric Perspective on Model Monitoring

Murtuza N Shergadwala,[1] Himabindu Lakkaraju,[2] Krishnaram Kenthapadi[1]

[1] Fiddler AI, Palo Alto, CA, USA
[2] Department of Computer Science, Harvard University, Cambridge, MA, USA
murtuza@fiddler.ai, hlakkaraju@hbs.edu, krishnaram@fiddler.ai

Abstract

Predictive models are increasingly used to make various consequential decisions in high-stakes domains such as healthcare, finance, and policy. It becomes critical to ensure that these models make accurate predictions, are robust to shifts in the data, do not rely on spurious features, and do not unduly discriminate against minority groups. To this end, several approaches spanning various areas such as explainability, fairness, and robustness have been proposed in recent literature. Such approaches need to be human-centered as they cater to the understanding of the models to their users. However, there is little to no research on *understanding the needs and challenges in monitoring deployed machine learning (ML) models from a human-centric perspective*. To address this gap, we conducted semi-structured interviews with 13 practitioners who are experienced with deploying ML models and engaging with customers spanning domains such as financial services, healthcare, hiring, online retail, computational advertising, and conversational assistants. We identified various human-centric challenges and requirements for model monitoring in real-world applications. Specifically, we found that relevant stakeholders would want model monitoring systems to provide clear, unambiguous, and easy-to-understand insights that are readily actionable. Furthermore, our study also revealed that stakeholders desire customization of model monitoring systems to cater to domain-specific use cases.

1 Introduction

Machine learning (ML) is increasingly playing an integral role in our day-to-day experiences. Increasingly, the applications of ML are no longer limited to search and recommendation systems, such as web search and movie and product recommendations, but ML is also being used in decisions and processes that are critical for individuals, businesses, and society. With ML based solutions and pipelines in high-stakes applications such as hiring, lending, criminal justice, healthcare, and education, the resulting personal and professional implications of ML are far-reaching. Consequently, it becomes critical to ensure that the underlying ML models are making accurate predictions, are robust to shifts in the data, are not relying on spurious features, and are not unduly discriminating against minority groups. This emerging field, called *model monitoring*, can be viewed as part of a broader ML model governance (Kurshan, Shen, and Chen 2020) and responsible ML framework (Arrieta et al. 2020), and is at an inflexion point, as evidenced by legal/regulatory requirements, requirements from the perspective of web-scale ML applications, and adoption of practical and scalable approaches. Model monitoring is receiving greater attention in light of regulations such as EU GDPR, CCPA, and the EU Trustworthy AI[1] initiative and several ML deployment failures in practice.

From an operational angle, large-scale ML systems require maintenance not only by the virtue of possessing software code but also because of the nuances of ML as a domain itself (Sculley et al. 2015). ML-specific nuances include dependency on data whose distributions can shift during production from when the model was designed and the dependency of a model on another model's output which can cascade issues from the other model onto the dependent model (Sculley et al. 2015). Such nuances imply that maintenance of ML systems requires *monitoring* its various aspects, such as data and models. Such monitoring is essential to ensure that the model does not become stale or degrade in performance due to changing real-world conditions or changes in the data collection process and data processing pipelines. In Figure 1, we illustrate the various activities in the lifecycle of an ML model and highlight the focus of this work in the monitoring activity. More broadly, the emerging field of *model monitoring* pertains to practices for deploying and maintaining ML models in production reliably and efficiently (Mäkinen et al. 2021). Monitoring of deployed ML models is needed to determine how often the model needs to be retrained and handle the following issues: (1) Data drift: The distribution of features may change over time, causing the quality of predictions made by the model to gradually degrade (Breck et al. 2019). (2) Changes in the relationships between input and target variables: The changes in real-world conditions may alter the relationships between input and target variables (often referred to as "concept drift" (Gama et al. 2014; Tsymbal 2004)), and thereby result in degraded model performance. (3) Data integrity and operational challenges: Compared to traditional software, ML models are more tolerant to unintended or unexpected

[1] https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

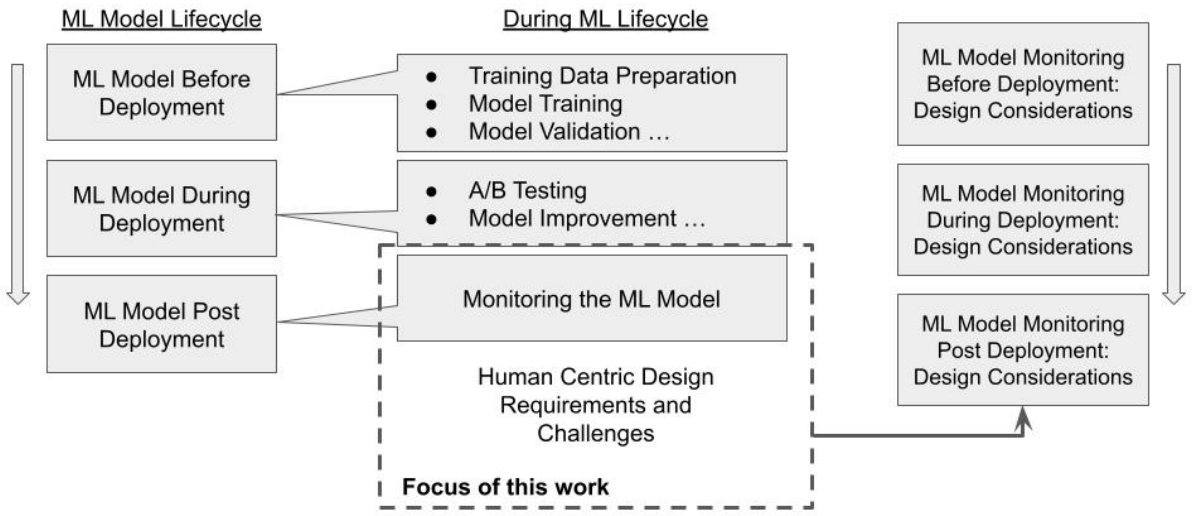

Figure 1: Various activities during ML model life cycle and the focus of this work.

changes in their inputs, and hence may continue to make predictions, even when the inputs may have been corrupted. As a result, the predictions may be erroneous or of poor quality. Thus, it is essential to ensure data integrity and detect any undesirable changes in the data pipeline (e.g., changing the measurement unit of a feature from feet to yards). (4) Reduced performance for subgroups of users: Although the model performance may not change as a whole, it is possible for the model to exhibit poor performance for certain subgroups of users (for whom, say, the relationship between input and target variables may have changed). Hence, in addition to monitoring overall accuracy and other performance measures, it is important to ensure that the model does not develop bias, and instead performs well across various subgroups of users.

We note that the above issues are of interest not only for data scientists but also for ML engineers, product managers, business decision makers, policy, compliance, and legal teams, internal and external auditors, and other stakeholders. In other words, given the importance of model monitoring as part of a broader AI model governance framework, model monitoring need to be human-centered not just in terms of usability by humans but also accounting for human behavior (Shneiderman 2021; Wing 2021). In this work, our goal is to understand the needs and challenges in monitoring deployed ML models from a human-centric perspective. This perspective is absolutely critical and is missing from existing literature.

Key Contributions: The goal of our study is to unearth practical and real-world challenges that are often encountered in real-world settings employing machine learning models. We conducted semi-structured interviews with 13 practitioners who are experienced with deploying ML models and engaging with customers spanning domains such as financial services, healthcare, hiring, online retail, computational advertising, and conversational assistants. We identified various human-centric challenges and requirements for model monitoring in real-world applications. Specifically, we found that relevant stakeholders would want model monitoring systems to provide clear, unambiguous, and easy-to-understand insights that are readily actionable. Furthermore, our study also revealed that stakeholders desire customization of model monitoring systems to cater to domain-specific use cases.

2 Related Work

This work lies at the intersection of several emerging areas of machine learning research, namely, detecting dataset shifts, monitoring model behavior via model understanding and explanations, monitoring fairness and robustness of machine learning models, and human-centered studies and open source tools focused on the aforementioned aspects. Below, we discuss some of the key works across each of the aforementioned areas.

Dataset Shifts There is a rich literature on techniques for detecting shifts in the data (e.g., see Breck et al. (2019); Cormode et al. (2021); Gama et al. (2014); Karnin, Lang, and Liberty (2016); Tsymbal (2004); Webb et al. (2016); Žliobaitė, Pechenizkiy, and Gama (2016) and the references therein). Both verifying the validity of model inputs and detecting changes in the features or model outputs are important challenges encountered in practical ML applications. The former is often addressed by including user-defined tests such as tests to check if a feature value is within a specified range (Schelter et al. 2018). For the latter, statistical hypothesis testing and confidence interval based approaches have been proposed. Statistical hypothesis testing involves checking if two given sets of samples are drawn from the same distribution by using a test statistic. Student's t-test and Kolmogorov–Smirnov test are examples of commonly used tests (Wasserman 2004; Murphy 2012). More advanced

tests such as Maximum Mean Discrepancy can also be used for higher dimensional data (Gretton et al. 2012). As these tests require sufficient fine-tuning (e.g., selecting the kernel and its hyperparameters), confidence interval based approaches (Efron and Tibshirani 1994) are often employed for detecting drifts in practice (Nigenda et al. 2022). In addition to the above approaches which are model-agnostic, specialized methods that leverage the model internals and the training data have also been proposed to determine the extent of drift and take remedial steps (Garg et al. 2020; Lipton, Wang, and Smola 2018; Reddi, Poczos, and Smola 2015; Wu et al. 2019).

Interpretability As argued by several recent works (Doshi-Velez and Kim 2017), model understanding is absolutely critical to ensure that ML models are relying on appropriate features when making predictions. To this end, model interpretations and explanations are widely being used to monitor model behavior. Many approaches have been proposed to directly learn interpretable models for various tasks including classification (Letham et al. 2015; Wang and Rudin 2015; Lakkaraju, Bach, and Leskovec 2016; Lou, Caruana, and Gehrke 2012; Bien and Tibshirani 2009) and clustering (Kim, Rudin, and Shah 2014; Lakkaraju and Leskovec 2016). To this end, various classes of models such as decision trees, decision lists (Letham et al. 2015), decision sets (Lakkaraju, Bach, and Leskovec 2016), prototype (case) based models (Bien and Tibshirani 2009), and generalized additive models (Lou, Caruana, and Gehrke 2012; Caruana et al. 2015) were proposed. However, complex models such as deep neural networks and random forests are often shown to achieve higher accuracy than simpler interpretable models (Ribeiro, Singh, and Guestrin 2016); thus, there has been a lot of interest in constructing post hoc explanations to understand their behavior.

A variety of post hoc explanation techniques have been proposed, which differ in their access to the complex model (i.e., black box vs. access to internals), scope of approximation (e.g., global vs. local), search technique (e.g., perturbation-based vs. gradient-based), explanation families (e.g., linear vs. non-linear), etc. For instance, LIME (Ribeiro, Singh, and Guestrin 2016) and SHAP (Lundberg and Lee 2017), are *model-agnostic*, *local explanation* approaches that explain individual predictions of any black box model by training a linear model locally around each prediction. These approaches rely on input perturbations to learn these interpretable local approximations. Several other *local explanation* methods have been proposed that compute *saliency maps* which capture importance of each feature for an individual prediction by computing the gradient with respect to the input (Simonyan, Vedaldi, and Zisserman 2014; Sundararajan, Taly, and Yan 2017; Selvaraju et al. 2017; Smilkov et al. 2017). A number of other local explanation methods (Koh and Liang 2017; Ribeiro, Singh, and Guestrin 2018) have also been proposed in the literature. An alternate approach is to provide a global explanation summarizing the black box as a whole (Lakkaraju et al. 2019; Bastani, Kim, and Bastani 2017), typically using an interpretable model.

Some recent work has shed light on the downsides of post hoc explanation techniques. Rudin (2019) argues that post hoc explanations are not reliable, as these explanations are not necessarily faithful to the underlying models and present correlations rather than information about the original computation. There has also been recent work on exploring vulnerabilities of black box explanations (Adebayo et al. 2018; Slack et al. 2020; Lakkaraju and Bastani 2020a; Rudin 2019; Dombrowski et al. 2019)—e.g., Ghorbani, Abid, and Zou (2019) demonstrated that post hoc explanations can be unstable, changing drastically even with small perturbations to inputs.

Fairness It is crucial to ensure that ML models deployed in real-world applications do not unduly discriminate against minority subgroups. To this end, monitoring the fairness of ML models has become common place in recent times (Dwork et al. 2012; Hardt, Price, and Srebro 2016). The initial literature on fairness in machine learning emphasized heavily on outlining the precise definitions of statistical fairness (Hardt, Price, and Srebro 2016). Several competing and contrasting notions of fairness emerged during this phase which can be broadly categorized into: 1) *group fairness* which emphasizes that protected groups should receive similar treatment as that of advantaged groups (Berk et al. 2018; Hardt, Price, and Srebro 2016) 2) *individual fairness* which requires that *similar* individuals to be treated similarly (Dwork et al. 2012), and 3) counterfactual fairness which captures the intuition that a decision pertaining to an individual is fair if it is the same in the actual world and a counterfactual world where the individual belonged to a different demographic group (Kusner et al. 2017). Furthermore, various metrics have been proposed to realize each of the aforementioned notions of fairness. For example, statistical (demographic) parity, equalized odds, equality of opportunity, and predictive parity are metrics proposed to enforce group fairness.

There are pros and cons to each of the aforementioned notions and metrics of fairness. For example, Dwork et al. (2012) argue that the group fairness notion of statistical parity leads to highly undesirable outcomes e.g., one might end up incarcerating women who pose no safety risk to ensure the same proportions of men and women are released. On the other hand, Kim, Reingold, and Rothblum (2018) highlight that assessing individual fairness is often hard in practice because it is hard to determine what is an appropriate metric function to measure the similarity of two individuals. Similarly, realizing counterfactual fairness in practice is also non-trivial because we do not have access to the counterfactuals of real world decisions i.e., there is no ground truth to determine if someone would have been incarcerated if that individual was a male instead of being a female and vice versa. Furthermore, prior research has also established that certain notions of fairness (calibration and balance conditions) are fundamentally incompatible and cannot be simultaneously optimized (Kleinberg, Mullainathan, and Raghavan 2017; Chouldechova 2017).

Open Source and Commercial Tools Several open source and commercial frameworks for monitoring deployed

ML models have been developed in recent times. Examples of such frameworks include Amazon SageMaker Model Monitor (Nigenda et al. 2022) & Clarify (Hardt et al. 2021), Deequ (Schelter et al. 2018), Evidently (Evidently 2022), Fiddler's Explainable Monitoring (Fiddler 2022), Google Vertex AI Model Monitoring (Taly, Sato, and Gruia 2021), IBM Watson OpenScale (IBM 2022), Microsoft Azure MLOps (Azure 2022), and Uber's Michelangelo platform (Hermann and Balso 2017). In contrast to these tools and techniques, the focus of this work is on understanding the needs, requirements, and challenges associated with monitoring deployed models from the perspective of relevant stakeholders.

Human-Centric Perspectives and User Studies There have been several user studies and interviews to understand the desiderata for model explanations and fairness (Doshi-Velez and Kim 2017; Cheng et al. 2021). For instance, (Bhatt et al. 2020) conducted interviews with data scientists to understand the use cases and accompanying desiderata for explaining models. On the other hand, Lakkaraju and Bastani (2020b) carried out a user study to understand if misleading explanations can fool domain experts into deploying racially biased models, while Kaur et al. (2020) found that explanations are often over-trusted and misused. Similarly, Poursabzi-Sangdeh et al. (2021) found that supposedly-interpretable models can lead to a decreased ability to detect and correct model mistakes, possibly due to information overload. Lage et al. (2019) used insights from rigorous human-subject experiments to inform the design of explanation algorithms. While the above works touch upon human-centric perspectives on explainability and fairness, they do not focus on human-centric perspectives on model monitoring which is the key goal of our work.

3 Study Design

We collected desiderata data for model monitoring from 13 practitioners who are experienced with deploying ML models and engaging with customers spanning domains such as financial services, healthcare, hiring, online retail, computational advertising, and conversational assistants. We did so by conducting semi-structured and one-on-one interviews virtually and analyzed the results from each interviewee. All the interviewees had an understanding about ML model monitoring and had interacted with various model monitoring tools.

In the following, we discuss the experimental design choices, the data collection process and the data analysis approach for the study.

3.1 Data Collection

We collected desiderata from thirteen (n=13) ML practitioners with expertise in the model monitoring space. We recruited these participants by reaching out to ML practitioners who are either working on developing model monitoring tools or needing such tools for their application use cases.

Based on an average of 2.7 years of professional experience of the interviewees within the MLOps space, we considered four to be experts, four to be beginners, and five to be intermediate. The interviews were semi-structured due to the exploratory and human-centered nature of the study which required prompting interviewees with follow-up questions relevant to their domain-specific experiences with model monitoring. The interview questions and their rationales are tabulated in Table 1.

Question	Motivation
IQ1: What kind of applications do you use ML models for?	To understand domain-specific use cases
IQ2: Why do you need model monitoring in these applications?	To understand domain-specific desiderata
IQ3: What aspects of model monitoring do you need?	To understand interpretations of model monitoring
IQ4: What would an ideal model monitoring framework look like and what do you want this framework to tell you?	To understand human-centered desiderata for model monitoring

Table 1: Interview Questions (IQ) for Model Monitoring in Practice

3.2 The Choice of Subject Pool: MLOps Practitioners

We chose ML practitioners working in the Machine Learning Operations (MLOps) space for several reasons. First, MLOps is a relatively new and emerging field where questions such as IQ3 in Table 1 are still utilized to understand people's perspectives on what constitutes as model monitoring as there is no set definition for it yet. Thus, MLOps practitioners are well versed with model monitoring related challenges as they face them in their day to day activities. Second, the MLOps practitioners we interviewed have experience with domain-specific use cases and would thus be able to better articulate fuzzy application and business requirements. Third, MLOps practitioners develop model monitoring solutions which enables them to discuss practical implementation challenges as well.

3.3 Data Analysis

We analyzed the responses to all the questions using inductive content analysis (Krippendorff 1980). The interview notes were analyzed in two ways. First, each transcript was individually annotated for model monitoring application areas/use cases, requirements, and challenges described by each of the interviewees. Second, the responses to each question mentioned in Table 1 across all interviews were pooled and analyzed to inductively generate common themes and categories for ML monitoring application areas, design requirements, and challenges.

Responses to IQ1 were identified either as a domain or a use case. For example, phrases and words such as "financial services", "insurance", "banking", and "adtech" were

labeled as domains. Whereas, phrases such as "fraud detection", "credit lending", and "speech recognition" were labeled as a use case. Responses to IQ2, IQ3, and IQ4 were analyzed to identify human-centric desiderata for model monitoring. Sentences that contained phrases such as "need to", "should have", "be able to", and "it would be great if" were labeled as requirements. Phrases such as "difficulty", "challenges", "not possible", "hard", and "risky" were analyzed to identify challenges faced by the interviewees. Further, Responses to IQ3 were also analyzed to discover interviewees' interpretation of model monitoring and what it entails. To do so, authors' domain knowledge of model monitoring was leveraged to label the aspect of model monitoring discussed by an interviewee. For example, if interviewees discussed data drift as a part of the response to IQ3, data drift was labeled as an aspect of model monitoring for that interviewee. The IQ3 labels across all interviewees were pooled to characterize various aspects of model monitoring as discussed by them.

4 Application Areas, Design Considerations and Challenges for Model Monitoring: Practitioners' Perspectives

In this section, we discuss the use cases, requirements, and challenges discussed by the interviewees. We begin with the results of IQ3 analysis, where we describe various aspects of model monitoring. Then, we discuss the results of IQ1 analysis, namely, the application areas mentioned by the interviewees. Then, we discuss the interviewees' desiderata and challenges for model monitoring.

4.1 What is Model Monitoring? Practitioners' Perspective

We analyzed the responses to IQ3 contextually to identify the key aspects of model monitoring stated by the interviewees. All the interviewees discussed *model performance monitoring* and *data drift monitoring* as a part of model monitoring. One interviewee discussed, *"we are interested in prediction drift as a proxy for model performance."* Another interviewee mentioned, *"Data scientists care more about model performance, deploying more models, etc."*. Similarly, one interviewee highlights their need for model monitoring by mentioning, *"we need tools for continuous model monitoring to assess end-to-end impact of these [design] changes and ensure model performance."*

Five (out of thirteen) interviewees emphasized *monitoring model fairness, bias, and model versions* as a part of model monitoring. One interviewee said, *"A subset of our models needs fairness analysis to ensure compliance."* Another interviewee mentioned their interaction with a banking client as follows: *"If you are a risk manager for banks, then you probably care about fairness/bias."* One interviewee discussed fairness monitoring as an aspect of model monitoring and mentioned, *"Monitoring fairness could be another interesting addition for my clients who are at the receiving end of regulatory constraints, fines, etc."* With respect to model versioning, an interviewee stated, *"At* [retracted organization name], *we have a champion model and a contender model. We used things such as A/B tests to evaluate both the models. [This is where] I think model monitoring can immensely help."*

We also noticed interviewees categorizing or grading the relative importance of various aspects of model monitoring as a part of the response to IQ3. Interviewees largely identify monitoring data integrity (Boritz 2005), that is, the accuracy, completeness, and consistency of data for inputs and outputs of a model as a *basic requirement*. One interviewee remarked, *"I think, at the very least, [we need] a system that monitors the exhaust of the model; the outputs and also the inputs of the model."* In addition to this, identification of data drift, performance drift, and outlier detection were mentioned as important but *intermediate requirements* for model monitoring. Another interviewee stated, *"On top of basic input output monitoring, another aspect is model performance from accuracy and precision point of view."* Finally, monitoring model fairness and bias were expressed as regulation driven, "good to have", and are thus labeled as *advanced requirements* for model monitoring.

4.2 Application Areas and Use Cases

Analyzing responses to IQ1 enabled us to identify specific domains and use cases of interest to the interviewees as shown in Table 2. Some example excerpts include: *"of course, classic are fraud, churn use cases, and recommendation for anything [in] retail."* Another interviewee states, *"our primary market is financial services and fintech, we have some retail, but insurance is clearly a new area where we are seeing new interest."* Regarding the financial services domain, one interviewee specifically mentioned that *"they're [financial services domain] the most advanced with the use of machine learning or they have the highest bar of regulatory scrutiny."* In the context of monitoring for online advertisement, one interviewee said, *"if by mistake we advertise to the wrong user, and by mistake, we bid with the wrong price, that will affect the end results, the revenues or the profit."* While there are numerous other domains and use cases for ML models, the results here are intended to contextualize the responses of the interviewees to the desiderata and the challenges discussed below.

Domain	Use Case
AdTech	Ads personalization and ads pricing.
Consumer Technology	Wake-word detection, automatic speech recognition, natural language understanding and interpretation, entity resolution, and text-to-speech generation.
Financial Services	Fraud detection, credit lending, and churn prediction.
Insurance	Risk prediction
Retail	Recommendation models and traffic monitoring.

Table 2: Domains and use cases discussed by the interviewees.

4.3 Human-centric Requirements for Model Monitoring

The following themes for model monitoring requirements emerged based on interviewees' responses to IQ2, IQ3, and IQ4.

Domain-Specific Debugging & Root Cause Analysis: Interviewees discussed the need for a model monitoring system to discover slices or sub-populations of data where unexpected model behavior and outcomes occur. This would help one gain insight on model errors, when to retrain a model, and domain-specific nuances. One interviewee mentioned, *"If it can provide me with some early warnings or signs where things go wrong and give me ways to resolve what's going on."* Further, the system should allow customizable levels of abstractions such as feature-level monitoring, prediction-level monitoring, and performance-level monitoring based on the use case. One interviewee stated, *"the thing that would really help is customizing monitoring by allowing different overlays of time scale, business metrics, model metrics, etc."*.

Risk Management, Model Governance, and Privacy: Interviewees would like model monitoring systems to help them manage risk and ensure regulatory compliance. Interviewees emphasized the need for a monitoring system to enable centralization of model governance in an organization rather than have dependencies on an individual or a team that created the model. One of the interviewees cited the work of (Kurshan, Shen, and Chen 2020) in discussing the challenges of model governance and mentioned the need for frameworks with self-regulatory capabilities. Such a system was also discussed to require to reduce human dependency and automate the process of error detection in ML pipelines. One of the interviewees remarked, *"In the past, people would manually go [look for errors] and document [the errors] but now we are trying to automate by monitoring."* Interviewees discussed the risk associated with retraining the model without monitoring. One interviewee said, *"[without monitoring] model refresh can be too risky. How do we ensure the new model is working as intended?"* In certain settings, it would also be desirable for the monitoring system to ensure that privacy and confidentiality of various assets such as protected user information in training data and intellectual property associated with the models are protected. One of the interviewees mentioned, *"that's [privacy] keeping people from not sending the data to a hosted environment; they want to keep it in their own private VPC and that adds a lot of challenges for monitoring [as a service]."*

Human-Centered Design: Interviewees discussed the need to preserve human autonomy and decision-making. In other words, the monitoring system should not trigger actions such as automated retraining, but instead provide actionable insights and suggestions to enable humans to make better decisions. Further, the monitoring system should provide relevant and meaningful alerts without cognitive overload. One interviewee discussed, *"For individual data scientists, I think the process [of root cause analysis] becomes cumbersome or if they are trying to share information, it becomes untenable when the number of models grows."* In the context of emotions, one interviewee stated, *"you know if somebody wakes me up at 12 o'clock in the night saying 'Oh, there is a drift and take a look,' and if there is no drift, I am going to be mad."* Another interviewee said, *"An ideal monitoring system should not create alert fatigue."* To avoid this, the model monitoring system needs to be aware of aspects such as the types of alerts, how often they are fired, and how they are presented. The challenge, however, as discussed by one interviewee is that, *"we may not have the right metrics that measure human [centered] things such as individual preferences and fatigue."*

4.4 Challenges for Model Monitoring: Temporal Categorization

We temporally categorized challenges as (1) design challenges before deploying a monitoring system, (2) challenges during monitoring an ML system, and (3) challenges post deployment with respect to the usefulness of monitoring outcomes.

Interviewees highlighted several *practical challenges in designing and deploying an ML model monitoring system*. These challenges included design questions such as what should be monitored, and how should the monitoring system interact with the model. One interviewee discussed, *"sometimes you have models that translate from one layer to another. So monitoring a feature XYZ may not make sense to the customer."* Another interviewee discussed that for monitoring systems as a service, *"when it comes to actually doing things, there are a lot of other constraints that become more important like people [clients] might not talk about [their use case] or might not explicitly tell you so."* These point to the requirement of a monitoring system having the ability to provide some guidelines or in-built options for what to monitor in an ML system.

Interviewees also discussed whether the monitoring system would also need the technical dependencies that a model requires, such as packages and modules, to ingest a model and execute successfully. One interviewee discussed, *"let's say an organization uses a package but there I'm not quite sure what the business model is, can anybody go and download that and set that up and run that on their end? I'm not sure."* Thus, monitoring systems are discussed as systems that have the ability to take an ML system as an input and provide monitoring insights as an output. In that context, interviewees raise the challenge for an ML monitoring system to possess a super-set of the technical capabilities of different ML systems. We note that this *perceived challenge* by the interviewees in practice is only a concern if the ML model is required to run specific predictions on the monitoring platform. We discuss this finding in Section 5.

Some interviewees had prior experience with model monitoring systems, and discussed the challenges of protecting privacy of users in training data and confidential information / intellectual property associated with their models. One interviewee said, *"For third party monitoring tools offered as SaaS [Software-as-a-Service], the clients have to send the data [to the third party server] – that becomes a privacy challenge."* All interviewees emphasized the challenges of adapting a monitoring system to the domain-specific needs

of the ML system and the lack of solutions that cater to their specific needs. For example, the data volume experienced by an ML system is highly sensitive to its application context, and could become a key design consideration for an ML monitoring system to be able to handle. Such decisions are currently made manually, and there is a lack of a framework that helps automate the design process for an ML monitoring system. Hence, interviewees highlighted the need for a fully managed monitoring service for their ML systems.

We also observed that the interviewees discussed *challenges that may occur when an ML monitoring system is deployed*. They discussed the lack of existing reliable solutions in assessing whether an observed drift in data or model performance is a cause for concern. One interviewee mentioned *"products have some sort of seasonality or recurring drift that is probably not very interesting for the developers to get alerted for. For example, when I was at [organization name] we would see usage peak on weekends vs. weekdays because most people would open the app on the weekends. If [organization name] were to use a [model] monitoring system, how do we identify drift that is anomalous vs. simply seasonal [or periodic]?"* Such a lack of reliability can result in cognitive fatigue that may desensitize practitioners from gaining meaningful insights from the monitoring system. Interviewees also discussed latency challenges, that is, how quickly can a monitoring system detect issues and suggest remedial actions. Here, interviewees discuss the need for the monitoring system to inform the human user about the computation time of various aspects of monitoring that may not necessarily be computed on similar timescale. One interviewee stated, *"I can do an F1 score in real time and that's a simple thing to be able to do, given a particular classification threshold but if I want to do AUC which is independent of the classification threshold, all of a sudden that's a very different computation. So it's very interesting that people know the formula, but they expect the same behavior for both which is not theoretically possible."* Systems that are slow to identify issues may not be useful in certain contexts, such as autonomous driving, where high-stakes decisions often need to be made in real-time by ML models. Further, due to privacy, intellectual property, and security/-compliance considerations, interviewees highlighted that the monitoring tools and systems may need to be present "on-premise" or in-situ rather than a third party housing such a system. Consequently, debugging or maintaining the monitoring system itself may be challenging, especially if the services are being sought from a third party.

Interviewees also discussed the *"so what" or value-based challenges* with monitoring systems. These discussions focused on the value of the insights gained from such systems, and pertained to aspects such as whether/how the monitoring insights could enable the stakeholders to take concrete actions to improve business outcomes and whether non-experts would be able to understand these insights. As an example, one interviewee quoted, *"developers need to know how the model affects business metrics, and monitoring is important for that."*

4.5 Challenges for Model Monitoring: Feature-specific Categorization

As noted earlier, data drift, outlier detection, data integrity violation, model performance, and bias/fairness are the key dimensions of model monitoring highlighted by the interviewees. Next, we discuss the specific desiderata themes within some of these dimensions.

For data drift, we found that all the interviewees consider input and output data distribution monitoring as a necessity for model monitoring. Also, such drift monitoring is treated as an indicator for model retraining. Further, data integrity violation and outlier detection are aspects that are discussed as a part of the data drift monitoring functionality as well. One interviewee said, *"customers want simple stuff like data drift, performance monitoring, and some sort of anomaly detection. That is the first set of things they want."* Another interviewee mentioned, *"customers are very worried about data integrity issues such as a breakdown in their pipeline they want to know immediately. A sudden change in data."* We note here that the "customers" referred to by the interviewees are stakeholders interacting with these interviewees in professional settings who have expressed interest in using model monitoring systems as a service.

For bias/fairness monitoring, customer requirements are currently driven through policies and regulations. One interviewee stated, *"Fairness is rare so far in my experience but with big banks, regulation and compliance are driving it."* Another interviewee said, *"There is a smaller subset of cases for bias and fairness monitoring."* Words such as "policy","regulatory constraints","fines", and "regulatory push" were used to describe the need for fairness monitoring. Practitioners mentioned that compliance and risk management teams are concerned about bias/fairness also due to its impact on the trust and the reputation of a company. By leveraging ML model monitoring, an organization could proactively detect and mitigate any biases observed in its deployed models instead of having to react when such issues are discovered by external entities.

5 Conclusion and Discussion

Motivated by the need for understanding the human-centered requirements and challenges in designing monitoring frameworks for ML systems, we performed an interview study with ML practitioners with experience spanning several application domains. We presented findings and insights on real-world use cases, desiderata, and challenges for ML model monitoring in practice based on these interviews. Interviewees discussed both feature-specific and process-specific aspects of model monitoring. Feature-specific aspects include monitoring data drift, model performance, and bias/fairness and ensuring that the alerts are relevant without cognitive overload. Process-specific aspects include the temporal considerations before, during, and after the deployment of the model monitoring system and the ability of a monitoring system to cater to different needs across the lifecycle of an ML system.

Based on the requirements and challenges reported in this work, we discuss potential pathways to address concerns on

themes of human-centric design of model monitoring systems. We believe that human autonomy and agency can be preserved using a human-AI decision making framework, wherein the model monitoring system is used in conjunction with a human decision maker. This implies enabling the system to report data integrity violations or different types of drift, and then allowing human users to pursue corrective pathways such as correction of labels or features, data re-sampling, and model retraining. To avoid cognitive load, monitoring systems could include threshold knobs and preference logging features that enable humans to embed domain-specific knowledge for their specific use cases. For centralization of model governance, the model monitoring system could enable automatic report generation and use of natural language to describe the state of an ML system. This would enable non-experts to also have an understanding of the state of an ML system. To address latency concerns, while there may be technological advancements to improve computation speed as well as efficient designs of a system, we also highlight the need for the system to educate the human user about the time estimates for certain computations as well as a comparative view of the difference in compute time for different aspects of a system. Such knowledge can potentially improve user experience while leveraging model monitoring systems.

We highlight a *perceived challenge* regarding the technical requirements of a model monitoring system. Interviewees described the challenge for an ML monitoring system to possess a super-set of the technical capabilities of different ML systems it monitors. However, we note that to monitor inputs, outputs, and other characteristics associated with a deployed model, a monitoring system does not need to execute the model and can instead take the production logs associated with the model as input. Thus, the model monitoring system may not need the technical infrastructure to run the model itself, and can instead focus on the *tools and techniques for prediction of distribution shifts*. This understanding influences the design decisions for monitoring systems and we thus highlight that model monitoring systems may be designed in a model-agnostic manner. This also points to the caution required in analyzing human-centric desiderata where perceived challenges by practitioners, who may not necessarily be experts in ML, may stem from misconceptions about the system functionalities.

We also note that inferring a user's cognition is currently an active area of research (Akula et al. 2022; Shergadwala, Panchal, and Bilionis 2022; Wu et al. 2022). While some human-centric requirements may point to the user's desire to be "understood" by a system in real-time, it is currently not practically feasible to so. Further, several studies have highlighted the limits of human oversight and the challenges that arise when attempting to build tools that enable people to monitor technological systems (e.g., see Perrow (1999) and Green (2022)). Thus, we need to be aware of the limits in the ability to build model monitoring tools and interfaces that actually satisfy the human-centric requirements stated by the interviewees.

We acknowledge that our analysis is limited based on the inputs of only thirteen practitioners. However, the interviewees are ML practitioners with deep knowledge of ML systems in their respective domains and extensive experience of ML model monitoring systems. Hence, we were able to obtain and analyze human-centric requirements and challenges from a practical viewpoint based on our interviews. More broadly, we encourage MLOps practices to formalize design frameworks for ML monitoring systems that are cautiously informed by human-centered desiderata.

Acknowledgments

We are thankful to the interviewees for their detailed responses to the questions. We also thank Joshua Rubin and Lea Genuit for their feedback and help with the analysis.

References

Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, 9505–9515.

Akula, A. R.; Wang, K.; Liu, C.; Saba-Sadiya, S.; Lu, H.; Todorovic, S.; Chai, J.; and Zhu, S.-C. 2022. CX-ToM: Counterfactual explanations with theory-of-mind for enhancing human trust in image recognition models. *Iscience*, 25(1): 103581.

Arrieta, A. B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information fusion*, 58: 82–115.

Azure. 2022. MLOps: Model management, deployment, lineage, and monitoring with Azure Machine Learning. https://tinyurl.com/57y8rrec. Accessed: 2022-02-02.

Bastani, O.; Kim, C.; and Bastani, H. 2017. Interpretability via model extraction. *arXiv preprint arXiv:1706.09773*.

Berk, R.; Heidari, H.; Jabbari, S.; Kearns, M.; and Roth, A. 2018. Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods & Research*.

Bhatt, U.; Xiang, A.; Sharma, S.; Weller, A.; Taly, A.; Jia, Y.; Ghosh, J.; Puri, R.; Moura, J. M.; and Eckersley, P. 2020. Explainable machine learning in deployment. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 648–657.

Bien, J.; and Tibshirani, R. 2009. Classification by set cover: The prototype vector machine. *arXiv preprint arXiv:0908.2284*.

Boritz, J. E. 2005. IS practitioners' views on core concepts of information integrity. *International Journal of Accounting Information Systems*, 6(4): 260–279.

Breck, E.; Polyzotis, N.; Roy, S.; Whang, S.; and Zinkevich, M. 2019. Data Validation for Machine Learning. In *MLSys*.

Caruana, R.; Lou, Y.; Gehrke, J.; Koch, P.; Sturm, M.; and Elhadad, N. 2015. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *Knowledge Discovery and Data Mining (KDD)*.

Cheng, H.-F.; Stapleton, L.; Wang, R.; Bullock, P.; Chouldechova, A.; Wu, Z. S. S.; and Zhu, H. 2021. Soliciting stakeholders' fairness notions in child maltreatment predictive systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–17.

Chouldechova, A. 2017. Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments. *Big Data*, 5(2): 153–163.

Cormode, G.; Karnin, Z.; Liberty, E.; Thaler, J.; and Veselỳ, P. 2021. Relative Error Streaming Quantiles. In *PODS*.

Dombrowski, A.-K.; Alber, M.; Anders, C.; Ackermann, M.; Müller, K.-R.; and Kessel, P. 2019. Explanations can be manipulated and geometry is to blame. *Advances in Neural Information Processing Systems*, 32.

Doshi-Velez, F.; and Kim, B. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science*, 214–226.

Efron, B.; and Tibshirani, R. J. 1994. *An introduction to the bootstrap*. CRC press.

Evidently. 2022. Evidently AI: Open-Source Machine Learning Monitoring. https://evidentlyai.com. Accessed: 2022-02.

Fiddler. 2022. Explainable Monitoring. https://www.fiddler.ai/ml-monitoring. Accessed: 2022-02.

Gama, J.; Žliobaitė, I.; Bifet, A.; Pechenizkiy, M.; and Bouchachia, A. 2014. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46(4): 1–37.

Garg, S.; Wu, Y.; Balakrishnan, S.; and Lipton, Z. 2020. A Unified View of Label Shift Estimation. In *NeurIPS*.

Ghorbani, A.; Abid, A.; and Zou, J. 2019. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3681–3688.

Green, B. 2022. The flaws of policies requiring human oversight of government algorithms. *Computer Law & Security Review*, 45.

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. 2012. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1): 723–773.

Hardt, M.; Chen, X.; Cheng, X.; Donini, M.; Gelman, J.; Gollaprolu, S.; He, J.; Larroy, P.; Liu, X.; McCarthy, N.; Rathi, A.; Rees, S.; Siva, A.; Tsai, E.; Vasist, K.; Yilmaz, P.; Zafar, M. B.; Das, S.; Haas, K.; Hill, T.; and Kenthapadi, K. 2021. Amazon SageMaker Clarify: Machine Learning Bias Detection and Explainability in the Cloud. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of Opportunity in Supervised Learning. In *30th Annual Conference on Neural Information Processing Systems 2016*, 3315–3323.

Hermann, J.; and Balso, M. D. 2017. Meet Michelangelo: Uber's Machine Learning Platform. https://eng.uber.com/michelangelo-machine-learning-platform. Accessed: 2022-02-02.

IBM. 2022. Validating and monitoring AI models with Watson OpenScale. https://tinyurl.com/5yzybu44. Accessed: 2022-02-02.

Karnin, Z.; Lang, K.; and Liberty, E. 2016. Optimal quantile approximation in streams. In *FOCS*.

Kaur, H.; Nori, H.; Jenkins, S.; Caruana, R.; Wallach, H.; and Wortman Vaughan, J. 2020. Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14.

Kim, B.; Rudin, C.; and Shah, J. A. 2014. The Bayesian Case Model: A Generative Approach for Case-Based Reasoning and Prototype Classification. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems 27*, 1952–1960. Curran Associates, Inc.

Kim, M.; Reingold, O.; and Rothblum, G. 2018. Fairness through computationally-bounded awareness. In *Advances in Neural Information Processing Systems*, 4842–4852.

Kleinberg, J.; Mullainathan, S.; and Raghavan, M. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In *Proceedings of the 8th Conference on Innovations in Theoretical Computer Science*, 43:1–43:23.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1885–1894. JMLR. org.

Krippendorff, K. 1980. *Content Analysis: An Introduction to Its Methodology (Commtext Series)*. Sage Publications, Thousand Oaks, California.

Kurshan, E.; Shen, H.; and Chen, J. 2020. Towards self-regulating AI: Challenges and opportunities of AI model governance in financial services. In *Proceedings of the First ACM International Conference on AI in Finance*.

Kusner, M. J.; Loftus, J.; Russell, C.; and Silva, R. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, 4066–4076.

Lage, I.; Chen, E.; He, J.; Narayanan, M.; Kim, B.; Gershman, S. J.; and Doshi-Velez, F. 2019. Human evaluation of models built for interpretability. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, 59–67.

Lakkaraju, H.; Bach, S. H.; and Leskovec, J. 2016. Interpretable Decision Sets: A Joint Framework for Description and Prediction. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 1675–1684.

Lakkaraju, H.; and Bastani, O. 2020a. "How do I fool you?": Manipulating User Trust via Misleading Black Box Explanations. In *AIES*.

Lakkaraju, H.; and Bastani, O. 2020b. "How do I fool you?" Manipulating User Trust via Misleading Black Box Explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 79–85.

Lakkaraju, H.; Kamar, E.; Caruana, R.; and Leskovec, J. 2019. Faithful and Customizable Explanations of Black Box Models. In *AAAI Conference on Artificial Intelligence, Ethics, and Society (AIES)*.

Lakkaraju, H.; and Leskovec, J. 2016. Confusions over time: An interpretable bayesian model to characterize trends in decision making. In *Advances in Neural Information Processing Systems (NeurIPS)*, 3261–3269.

Letham, B.; Rudin, C.; McCormick, T. H.; and Madigan, D. 2015. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics*.

Lipton, Z.; Wang, Y.-X.; and Smola, A. 2018. Detecting and correcting for label shift with black box predictors. In *ICML*.

Lou, Y.; Caruana, R.; and Gehrke, J. 2012. Intelligible models for classification and regression. In *ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, 150–158. ACM.

Lundberg, S. M.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Neural Information Processing Systems (NIPS)*, 4765–4774. Curran Associates, Inc.

Mäkinen, S.; Skogström, H.; Laaksonen, E.; and Mikkonen, T. 2021. Who Needs MLOps: What Data Scientists Seek to Accomplish and How Can MLOps Help? In *2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*, 109–112. IEEE.

Murphy, K. P. 2012. *Machine learning: A probabilistic perspective*. MIT press.

Nigenda, D.; Karnin, Z.; Zafar, M. B.; Ramesha, R.; Tan, A.; Donini, M.; and Kenthapadi, K. 2022. Amazon SageMaker Model Monitor: A System for Real-Time Insights into Deployed Machine Learning Models. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Perrow, C. 1999. *Normal Accidents: Living with High-Risk Technologies*. Princeton University Press.

Poursabzi-Sangdeh, F.; Goldstein, D. G.; Hofman, J. M.; Wortman Vaughan, J. W.; and Wallach, H. 2021. Manipulating and measuring model interpretability. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, 1–52.

Reddi, S.; Poczos, B.; and Smola, A. 2015. Doubly robust covariate shift correction. In *AAAI*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Knowledge Discovery and Data Mining (KDD)*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Rudin, C. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5): 206.

Schelter, S.; Lange, D.; Schmidt, P.; Celikel, M.; Biessmann, F.; and Grafberger, A. 2018. Automating large-scale data quality verification. In *VLDB*.

Sculley, D.; Holt, G.; Golovin, D.; Davydov, E.; Phillips, T.; Ebner, D.; Chaudhary, V.; Young, M.; Crespo, J.-F.; and Dennison, D. 2015. Hidden technical debt in machine learning systems. In *NeurIPS*.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Shergadwala, M. N.; Panchal, J. H.; and Bilionis, I. 2022. How Does Past Performance of Competitors Influence Designers' Cognition, Behaviors, and Outcomes? *Journal of Mechanical Design*, 144(10): 101401.

Shneiderman, B. 2021. Responsible AI: Bridging from ethics to practice. *Communications of the ACM*, 64(8): 32–35.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *International Conference on Learning Representations (ICLR)*.

Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; and Lakkaraju, H. 2020. How can we fool LIME and SHAP? Adversarial Attacks on Post hoc Explanation Methods.

Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F. B.; and Wattenberg, M. 2017. SmoothGrad: removing noise by adding noise. In *ICML Workshop on Visualization for Deep Learning*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International Conference on Machine Learning (ICML)*.

Taly, A.; Sato, K.; and Gruia, C. 2021. Monitoring feature attributions: How Google saved one of the largest ML services in trouble. Google Cloud Blog.

Tsymbal, A. 2004. The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 106(2): 58.

Wang, F.; and Rudin, C. 2015. Falling Rule Lists. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*.

Wasserman, L. 2004. *All of statistics: A concise course in statistical inference*, volume 26. Springer.

Webb, G. I.; Hyde, R.; Cao, H.; Nguyen, H. L.; and Petitjean, F. 2016. Characterizing concept drift. *Data Mining and Knowledge Discovery*, 30(4): 964–994.

Wing, J. M. 2021. Trustworthy AI. *Communications of the ACM*, 64(10): 64–71.

Wu, E. Q.; Cao, Z.; Sun, P. Z.; Li, D.; Law, R.; Xu, X.; Zhu, L.-M.; and Yu, M. 2022. Inferring Cognitive State of Pilot's Brain Under Different Maneuvers During Flight. *IEEE Transactions on Intelligent Transportation Systems*.

Wu, Y.; Winston, E.; Kaushik, D.; and Lipton, Z. 2019. Domain adaptation with asymmetrically-relaxed distribution alignment. In *ICML*.

Žliobaitė, I.; Pechenizkiy, M.; and Gama, J. 2016. An overview of concept drift applications. *Big data analysis: new algorithms for a new society*, 91–114.