Evaluation Gaps in Machine Learning Practice

Ben Hutchinson
Google Research
Sydney, Australia
benhutch@google.com

Negar Rostamzadeh
Google Research
Montreal, Canada
nrostamzadeh@google.com

Christina Greer
Google Research
Mountain View, USA
heller@google.com

Katherine Heller
Google Research
Mountain View, USA

Vinodkumar Prabhakaran
Google Research
San Francisco, USA

ABSTRACT

Forming a reliable judgement of a machine learning (ML) model’s appropriateness for an application ecosystem is critical for its responsible use, and requires considering a broad range of factors including harms, benefits, and responsibilities. In practice, however, evaluations of ML models frequently focus on only a narrow range of decontextualized predictive behaviours. We examine the evaluation gaps between the idealized breadth of evaluation concerns and the observed narrow focus of actual evaluations. Through an empirical study of papers from recent high-profile conferences in the Computer Vision and Natural Language Processing communities, we demonstrate a general focus on a handful of evaluation methods. By considering the metrics and test data distributions used in these methods, we draw attention to which properties of models are centered in the field, revealing the properties that are frequently neglected or sidelined during evaluation. By studying these properties, we demonstrate the machine learning discipline’s implicit assumption of a range of commitments which have normative impacts; these include commitments to consequentialism, abstractability from context, the quantifiability of impacts, the limited role of model inputs in evaluation, and the equivalence of different failure modes. Shedding light on these assumptions enables us to question their appropriateness for ML system contexts, pointing the way towards more contextualized evaluation methodologies for robustly examining the trustworthiness of ML models.

CCS CONCEPTS

• Computing methodologies → Machine learning approaches.

KEYWORDS

machine learning, evaluation, applications

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ACM Reference Format:

Ben Hutchinson, Negar Rostamzadeh, Christina Greer, Katherine Heller, and Vinodkumar Prabhakaran. 2022. Evaluation Gaps in Machine Learning Practice. In 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT ’22), June 21–24, 2022, Seoul, Republic of Korea. ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3531146.3533233

1 INTRODUCTION

When evaluating a machine learning (ML) model for real-world uses, two fundamental questions arise: Is this ML model good (enough)? and Is this ML model better than some alternative? Obtaining reliable answers to these questions can be consequential for safety, fairness, and justice concerns in the deployment ecosystems. To address such questions, model evaluations use a variety of methods, and in doing so make technical and normative assumptions that are not always explicit. These implicit assumptions can obscure the presence of epistemic gaps and motivations in the model evaluations, which, if not identified, constitute risky unknown unknowns.

Recent scholarship has critiqued the ML community’s evaluation practices, focusing on the use of evaluation benchmarks and leaderboards. Although leaderboards support the need of the discipline to iteratively optimize for accuracy, they neglect concerns such as inference latency, robustness, and externalities [51]. The structural incentives of the “competition mindset” encouraged by leaderboards can pose challenges to empirical rigor [154]. For example, over-reliance on a small number of evaluation metrics can lead to gaming the metric (cf. Goodhart’s Law “when a measure becomes a target, it ceases to be a good measure”) [162]; this can happen unintentionally as researchers pursue models with “state of the art” performance. Benchmarks that encourage narrowly optimizing for test set accuracy can also lead to models relying on spurious signals [31], while neglecting the challenge of measuring the full range of likely harms [22]. Birhane et al. find evidence for this in their study of the discourse of ML papers, showing that the field centers accuracy, generalization, and novelty, while marginalizing values such as safety [18]. Given that benchmark evaluations serve as proxies for performance on underlying abstract tasks [152], evaluating against a range of diverse benchmarks for each task might help mitigate biases within each benchmark. However, ML research disciplines seem to be trending towards relying on fewer evaluation benchmark datasets [94], with test set reuse potentially leading to a research community’s overfitting with respect to the general task [104, 178]. Furthermore, within each benchmark, items are weighted equally (thus focusing on the head of the data distribution), failing to capture inherent differences in difficulty across items, and hence providing poor measures of progress on task performance [142]. As Raji et al. point out, the ML
research discipline’s decontextualized and non-systematic use of benchmark data raises serious issues with regards to the validity of benchmarks as measures of progress on general task performance [136].

This paper complements and extends this range of critiques, considering the risks of application developers adopting the ML research community’s standard evaluation methodologies. We seek to address challenges in measuring technology readiness (TRAM) [105, 141], while acknowledging this cannot be reduced to a purely technical question [43, 141]. By studying and analyzing the ML research community’s evaluation practices, we draw attention to the evaluation gaps between ideal theories of evaluation and what is observed in ML research. By considering aspects of evaluation data and evaluation metrics—as well as considerations of evaluation practices such as error analysis and reporting of error bars—we highlight the discrepancies between the model quality signals reported by the research community and what is relevant to real-world model use. Our framework for analyzing the gaps builds upon and complements other streams of work on ML evaluation practices, including addressing distribution shifts between development data and application data [34, 95, 161], and robustness to perturbations in test items [76, 119, 133, 174]. We situate this work alongside studies of the appropriateness of ML evaluation metrics (e.g., [47, 89, 178]), noting that reliable choice of metric is often hampered by unclear goals [44, 98]. In foregrounding the information needs of application developers, we are also aligned with calls for transparent reporting of ML model evaluations [118], prioritizing needs of ML fairness practitioners [75], model auditing practices [137], and robust practices for evaluating ML systems for production readiness [23].

In Section 2, we consider various ideal goals that motivate why ML models are evaluated, discussing how these goals can differ between research contexts and application contexts. We then report in Section 3 on an empirical study into how machine learning research communities report model evaluations. By comparing the ideal goals of evaluation with the observed evaluation trends in our study, we highlight in Section 4 the evaluation gaps that present challenges to evaluations being good proxies for what application developers really care about. We identify six implicit evaluation assumptions that could account for the presence of these gaps. Finally, in Section 5, we discuss various techniques and methodologies that may help to mitigate these gaps.

2 IDEALS OF ML MODEL EVALUATION

Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise. — John Tukey [163, pp. 13–14]

Although this paper is ultimately concerned with practical information needs when evaluating ML models for use in applications, it is useful to first step back and consider the ultimate motivations and goals of model evaluation. To evaluate is to form a judgement; however, asking Is this a good ML model? is akin to asking such a question of other artefacts—such as Is this a good glass?—in that it requires acknowledging the implicit semantic arguments of uses and goals [135]. For example, Is this a good glass [for my toddler to drink from, given that I want to avoid broken glass]? is a very different question from Is this a good glass [in which to serve wine to my boss, given that I want to impress them]?.

In this paper, we will speak of a model evaluation as a system of arbitrary structure that takes a model as an input and produces outputs of some form to judge the model. Designing a model evaluation often involves choosing one or more evaluation metrics (such as accuracy) combined with a choice of test data. The evaluation might be motivated by various stakeholder perspectives and interests [92]. The output might, for example, produce a single metric and an associated numeric value, or a table of such metrics and values; it might include confidence intervals and significance tests on metric values; and it might include text. By producing such an output, the evaluation helps to enable transparency by reducing the number of both unknown unknowns and known unknowns.

For the purposes of this paper, it is useful to distinguish between two types of evaluations:

Learner-centric. An ML model evaluation system useful for evaluating the learner (i.e., machine learning algorithm).

Application-centric. An ML model evaluation system useful for evaluating a potential application.

Learner-centric evaluations make conclusions about the quality of the learner or its environment based on the evaluation of the learned model. These including evaluations motivated by novel learning algorithms or model architectures, but also ones that aim to shed light on the training data (for example ML model evaluations can shed light on the data-generating practices used by institutions [5]), or b) “Green AI” explorations of how the learner can efficiently use limited amounts of resources [153]. However, when we evaluate a model without a specific application in mind, we lose the opportunity to form judgements specific to a use case. On the other hand, application-centric evaluations are concerned with how the model will operate within an ecosystem consisting of both human agents and technical components (Figure 1), sometimes described as the “ecological validity” [46]. Applications often use scores output by the model to initiate discrete actions or decisions, by applying a specific classification threshold to the scores. In contrast, learner-centric evaluations sometimes care about scores output by models even in the absence of any thresholds.

This distinction between learner-centric and application-centric is related (albeit imperfectly) to the different objectives of model evaluations that concern the engineering and science disciplines [114, 169]. Note that we are not claiming (cf. the debate in [123]) that science lies outside the bounds of statistical/ML methods, but rather that scientific-flavored pursuits have distinct uses and goals [137]. For example, Diana Forsythe’s studies of 1980s AI labs [56]. Important to this debate regarding the scientific goals of ML is the question of construct validity; that is, whether our measurements actually measure the things that we claim they do [86, 87, 136]. Conversely, consequential validity—which includes the real-world consequences of an evaluation’s interpretation and use—is likely

---

[1] The history of this type of use case extends beyond ML models, e.g., to the use of regression models in university admissions testing [68].
Evaluation Gaps in Machine Learning Practice

Figure 1: Learner-centric ML model evaluations are concerned with the learner and its environment. Application-centric model evaluations are concerned with how the model will interact with an ecosystem into which it is introduced.

more important to considerations of accountability and governance of ML models in applications [87].

Scientific goal. Evaluating the model can motivate beliefs/explanations about the world (including possibly the learner).

Engineering goal. Evaluating the model can tell us whether the model can be used as a means towards a goal.

This distinction is closely related to one between “scientific testing” and “competitive testing” made by Hooker in 1995, who takes the position that competitive testing a) is unscientific, and b) does not constitute true research but merely development [79]. However, since engineering research has its own goals, distinct from those of science [26], a more defensible position is that evaluations in support of scientific research are distinct from evaluations in support of engineering research.

Table 1 summarizes the above distinctions and the relationships between them. The distinction between learner-centric and application-centric evaluations relates to the question of internal validity and external validity that is more commonly discussed in the social sciences than in ML (see, e.g., [124]) but also sometimes in ML [104]. This is reflected in the ways in which practitioners of the two types of evaluations discuss the topic of robustness. Learner-centric evaluations pay attention to the robustness of the learner to changes in the training data (e.g., distributional shifts, outliers, perturbations, poisoning attacks; and with connections to robust estimation of statistics [102]), while application-centric evaluations pay attention to desired behaviors such as the (in)sensitivity of the model to certain classes of perturbations of the input, or to sensitive input features (e.g., [61]).

Note that nothing in the ideals of evaluation described above has stipulated whether evaluations are quantitative or qualitative. For example, one could imagine interrogating a chatbot model using qualitative techniques, or adopting methodologies of political critique such as [41]. Similarly, nothing has stipulated what combinations of empirical or deductive methods are used.

3 ML MODEL EVALUATIONS IN PRACTICE

Beneath the technical issues lie some differences in values concerning not only the meaning but also the relative merit of “science” and “artificial intelligence.” — Diana Forsythe [56]

To shed light on the ML research community’s norms and values around model evaluation, we looked at how these communities report their model evaluations. By examining 200 papers from several top conferences in two research disciplines that use ML approaches extensively, we identified patterns regarding choices of metrics, evaluation data, and measurement practices. This empirical study of ML research practices complements several recent studies of ML evaluation practices. These include: a survey 144 research papers studying the properties of models that are tested for [178]; a review of 107 papers from Computer Vision (CV), Natural Language Processing (NLP) and other ML disciplines to diagnose internal and external modes of evaluation failures [104]; an analysis of whether 60 NLP and CV papers pay attention to accuracy or efficiency [153]; and an analysis of the Papers With Code dataset for patterns of benchmark dataset creation and re-use [94].

3.1 Method

3.1.1 Data. We sampled 200 research papers, stratified by discipline, conference and year. 100 papers were selected from each of the NLP and CV disciplines. We selected 20 papers at random from the proceedings of each of the 55th to 59th Annual Meetings of the Association of Computational Linguistics (ACL’2017–ACL’2021), 25 papers from each of the proceedings of the 2019–2021 IEEE Conferences on Computer Vision and Pattern Recognition (CVPR’2019–CVPR’2021), and 25 papers from the 24th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI’2021). These conferences represent the pinnacles of their respective research fields.3

3https://paperswithcode.com
3ACL and CVPR are rated A (“flagship conference”), and MICCAI is rated A (“excellent conference”), by core.edu.au; all three are in the top 30 computer science conferences out of over 900 listed on research.com.
3.1.2 Analysis. The authors of this paper performed this analysis, dividing the papers among themselves based on disciplinary familiarity. Using an iterative procedure of analysis and discussion, we converged on a set of labels that captured important aspects of evaluations across and within disciplines. Recall from Section 2 that, for our purposes, a single evaluation typically involves choosing one or more metrics and one or more datasets. We coded each of the papers along three dimensions. a) Metrics: Which evaluation metrics were reported? After iteration, we converged on the categories of metrics shown in Table 2. b) Data: Was test data drawn from the same distribution as the training data, under the Independent and Identically Distributed (I.I.D.) assumption? c) Analysis: Was statistical significance of differences reported? Were error bars and/or confidence intervals reported? Was error analysis performed? Were examples of model performance provided to complement measurements with qualitative information?

3.2 Results

Although each of the disciplines and conferences does not define itself solely in terms of ML, the practice of reporting one or more model evaluations in a research paper is ubiquitous. Only five papers did not include evaluations of ML models; of these two were published at ACL (a survey paper, a paper aimed at understanding linguistic features, and one on spanning-tree algorithms), and two at CVPR (a paper with only qualitative results, and one introducing a dataset). Table 3 summarizes the results of the other 195. Counts are non-exclusive, for example papers frequently reported multiple metrics and sometimes reported performance both on I.I.D. test data and on non-I.I.D. test data.

Appendix B contains an overview of the flavors of test data we observed. We found evidence to support the claim that evaluations of NLP models have “historically involved reporting the performance (generally meaning the accuracy) of the model on a specific held-out [i.e., I.I.D.] test set” [20, p. 94]. CV evaluations seem to be even more likely to utilize I.I.D. test data, and—consistent with [94]—CV papers typically either introduce a new task (and corresponding benchmark dataset) [88, 103, 143, 175] or present results of a new model on an existing widely-used benchmark [73, 139]. An exception to this trend was CV papers which explored shared representations (e.g., in multi-task learning [53, 100] or domain adaptation [120, 127]). Evaluations in both disciplines showed a heavy reliance on reporting point estimates of metrics, with variance or error bars typically not reported in our sample. While colloquial uses of phrases like “significantly better” were fairly common, most papers did not report on technical calculations of statistical differences; we considered only those latter instances when coding whether a paper reported significance. Regarding metrics, most of those that were frequently seen in our sample were somewhat insensitive to different types of errors. For example, accuracy does not distinguish between FP and FN; \( F_1 \) is symmetric in FP and FN (they can be swapped without affecting \( F_1 \)); the \( \text{Overlap} \) metrics are similar invariant to swapping of the predicted bounding box and the reference bounding box; the \( \text{Distance} \) category of metrics does not distinguish over-estimation from under-estimation on regression tasks.

From our reading of the 200 papers in our sample, one qualitative observation we had was that model evaluations typically do not include concrete examples of model behavior, nor analyses of errors (for a counterexample which includes these practices, see [35]). Also, we noted the scarcity of papers whose sole contribution is a new dataset for an existing task, aligning with previous observations that dataset contributions are not valued highly within the community [148]. We hypothesise that conference reviewers place emphasis on novelty of model, task, and/or metric. We note a general tension between disciplinary values of task novelty and demonstrating state-of-the-art performance by outperforming previous models, and the risk of overfitting from test set re-use discussed by [104].

### 3.3 Discussion

This small-scale quantitative study of model evaluations provides clues as to the values and goals of the ML research communities. Test data was often old (e.g., the CONLL 2003 English NER dataset [150] used in two papers); optimizing for these static test sets fails to account for societal and linguistic change [14]. Disaggregation of metrics was rare, and fairness analyses were absent despite our sample being from 2017 onward, concurrent with mainstream awareness of ML fairness concerns. Despite being acknowledged by influential thought-leaders in ML to be unrealistic for applications [15], using I.I.D. test data is the norm. These are in alignment with the learner-centric goals of evaluations (Section 2). Similarly, with a few exceptions in our sample, there was general paucity of discussions of tradeoffs such as accuracy vs resource-efficiency that are typical of engineering disciplines [26], suggesting that the ML research disciplines generally aspire to scientific goals concerning understanding and explaining the learner. With this lens, the disciplinary paradigm of measuring accuracy on I.I.D. test data is not surprising: the goal is to assess a model’s ability to generalize. This assessment would then give us good guarantees on the application’s behavior, if the practical challenges of ascertaining the data distributions in an application ecosystem can be overcome. In practice, however,
Table 2: Categories of evaluation metrics used in the analysis of the ML research literature. TP=true positives; TN=true negatives; FP=false positives; FN=false negatives; N=total number of data points. See Appendix A for the most common metrics in our data and their categorizations.

| Metric category | Examples | Description |
|-----------------|----------|-------------|
| **ACCURACY**    | Accuracy, error rate | Sensitive to the sum TP+TN and to N. Not sensitive to class imbalance. |
| **PRECISION**   | Precision, BLEU | Sensitive to TP and FP. Not sensitive to FN or TN. |
| **RECALL**      | Recall, Rouge | Sensitive to TP and FN. Not sensitive to FP or TN. |
| **F-score**     | F₁, Fβ | Sensitive to TP, FP and FN. Not sensitive to TN. |
| **OVERLAP**     | Dice, IoU | Sensitive to intersection and overlap of predicted and actual. |
| **LIKELIHOOD**  | Perplexity | Sensitive to the probability that the model assigns to the test data. |
| **DISTANCE**    | MSE, MAE, RMSE, CD | Sensitive to the distance between the prediction and the actual value. |
| **CORRELATION** | Pearson’s r, Spearman’s ρ | Sensitive to each of TP, TN, FP and FN, but unlike Accuracy metrics they factor in the degree of agreement that would be expected by chance. |
| **AUC**         | MAP, AUROC | Does not rely on a specific classification threshold, but instead calculates the area under a curve parameterized by different thresholds. |

Table 3: Analysis of how Natural Language Processing (NLP) and Computer Vision (CV) research communities perform ML model evaluations. ♣ Appendix A provides definitions of commonly observed metrics, and their mappings to categories. ♠ Includes any form of error bars/confidence intervals/credible intervals/variation across multiple runs. ♠ Report together here due to the equivalence of the Dice measure (in the OVERLAP category) and F₁ (in the F-score category) [128].

| Discipline/Venue  | NLP:ACL (97) | CV:CVPR (73) | CV:MICCAI (25) | CV:Combined (98) | NLP+CV:Combined (195) |
|-------------------|--------------|--------------|----------------|------------------|---------------------|
| **Most Common Metrics** |             |              |                |                  |                     |
| Metric category   | Accuracy (47) | AUC (32)     | Distance (14)  | AUC (38)         | Accuracy (76)       |
| (num. of papers)  |              |              |                |                  |                     |
| **PRECISION**     | F-score (45)  | Accuracy (25)| Overlap (9)    | Overlap (31)     | F-score + Overlap ♠ (74) |
| **RECALL**        | Precision (43)| Overlap (22) | AUC (6)        | Accuracy (29)    | Precision (48)      |
|                   | Recall (25)   | Distance (10)| AUC (4)        | Distance (24)    |                     |
| **CORRELATION**   | Pearson’s r   |              |                |                  |                     |
|                   | Spearman’s ρ |              |                |                  |                     |

| Data              | I.I.D. test data | Non-I.I.D. test data | | | |
|-------------------|-----------------|----------------------| | | |
| Reports significance | 24 | 0 | 7 | 7 | 31 |
| Reports error bars ♠ | 10 | 6 | 10 | 16 | 26 |

4 GAPS AND ASSUMPTIONS IN COMMON EVALUATION PRACTICES

In theory there is no difference between theory and practice, while in practice there is. — Brewster (1881) [25]

We now consider whether the research evaluation practices observed in Section 3 are aligned with the needs of decision-makers who consider whether to use a model in an application. That is, we consider whether the typically learner-centric evaluations, which commonly use metrics such as accuracy or F₁ on test data I.I.D. with the training data, meet the need of application-centric evaluations. In doing so, we expose, in a novel way, the interplay of technical and normative considerations in model evaluation methodologies.

4.1 Assumptions in Model Evaluation

We introduce six assumptions in turn, describing both how they operate individually in evaluations and how they compose and compound. We also call out “evaluation gaps” of concern relevant to each assumption. Appendix C contains a hypothetical example from a specific application domain that illustrates the flavors of the concerns. Our starting point is the observation from Section 2 that the goal of application-centric model evaluations is to understand how a model will interact with its ecosystem, which we denote schematically as:

Understand(Model + Ecosystem)

(Application-centric Evaluation Goal)

Assumption 1: Consequentialism. Consequentialism is the view that whether actions are good or bad depends only on their consequences [158]. The ML research literature often appeals to motivations about model utility to humans (e.g., [19, 27, 51, 59, 77, ...]

these challenges can be severe, and the research papers we surveyed do not generally tackle questions of uncertainty regarding data distributions.
including papers on fairness in ML such as [29, 36, 38, 39]). In adopting consequentialism as its de facto ethical framework, ML prioritizes the greatest good for the greatest number [85] and centers measurable future impacts. Moreover, the consequences that are centered are the direct consequences, with little attention given to motives, rules, or public acceptance [158]. This is realized as a focus on the first-order consequences of introducing the model into the ecosystem. Changes to the ecosystem itself—e.g., addressing what social change is perceived as possible and desirable [49, 68, 80]—are assumed to be out of scope, as are concerns for setting of precedents for other ML developers. We denote this assumption schematically as:

\[
\text{Understand} (\text{Model + Ecosystem}) \approx \text{Utility} (\text{Model + Ecosystem}) \quad \text{(Consequentialism Assumption)}
\]

**Evaluation Gap 1: Provenance.** A focus on future consequences neglects important moral considerations regarding the construction of the model. This excludes both deontological concerns—for example, Were data consent and sovereignty handled appropriately? [4, 41, 97] and Were data workers treated with dignity? [67]—as well as questions regarding past costs of development—for example, What were the energy use externalities of model training? [40, 60, 160] and Was the labour paid fairly? [157]. Schwartz et al. coin the phrase “Red AI” to describe ML work that disregards the costs of training, noting that such work inhibits discussions of when costs might outweigh benefits [153].

**Evaluation Gap 2: Social Responsibilities.** Another outcome of focusing primarily on direct consequences is marginalizing the assessment of a model against the social contracts that guide the ecosystem in which the model is used, such as moral values, principles, laws, and social expectations. For instance, Does the model adhere to the moral duty to treat people in ways that upholds their basic human rights? [156], Does it abide by legal mechanisms of accountability? [115, 138], and Does it satisfy social expectations of inclusion, such as the “nothing about us without us” principle? [33].

**Assumption 2: Abstractability from Context.** The model’s ecosystem is reduced to a set of considerations (X, Y), i.e., the inputs to the model and the “ground truth,” and in practice X may often fail to model socially important yet sensitive aspects of the environment [5, 10]. The model itself is reduced to a predicted value Ŷ, ignoring e.g., secondary model outputs such as confidence scores, or predictions on auxiliary model heads.

\[
\text{Utility} (\text{Model + Ecosystem}) \approx \text{Utility} (\hat{Y}, X, Y) \quad \text{(Assumption of Abstractability of Context)}
\]

**Evaluation Gap 3: System Considerations.** Equating a model with its prediction overlooks the potential usefulness of model interpretability and explainability. Also, reducing an ecosystem to model inputs and “ground truth” overlooks questions of system dynamics [112, 155], such as feedback loops, “humans-in-the-loop,” and other effects “due to actions of various agents changing the world” [15]. Also overlooked are inference-time externalities of energy use [28, 60], cultural aspects of the ecosystem [146], and long term impacts [29].

**Evaluation Gap 4: Interpretive Epistemics.** By positing a variable Y = y which represents the “ground truth” of a situation—even in situations involving social phenomena—a positivist stance on knowledge is implicitly adopted. That is, a “true” value Y = y is taken to be objectively singular and knowable. This contrasts with anthropology’s understanding of knowledge as socially and culturally dependent [57] and requiring interpretation [63]. In the specific cases of CV and NLP discussed in Section 3, cultural aspects of image and language interpretation are typically marginalized (cf. [11, 16, 90, 101], for example), exemplifying what Aroyo and Welty call AI’s myth of “One Truth” [7]. Furthermore, the positivist stance downplays the importance of questions of construct validity and reliability [58, 87].

**Assumption 3: Input Myopia.** Once the input variable X has been used by the model to calculate the model prediction Ŷ, X is typically ignored for the remainder of the evaluation. That is, the utility of the model is assumed to depend only on the model’s prediction and on the “ground truth.” We illustrate this with a causal graph diagram in Figure 2, which shows Utility as independent of X once the effects of Ŷ and Y are taken into account.

\[
\text{Utility} (\hat{Y}, X, Y) \approx \text{Utility} (\hat{Y}, Y) \quad \text{(Input Myopia Assumption)}
\]

**Evaluation Gap 5: Disaggregated Analyses.** By reducing the variables of interest to the evaluation to the prediction Ŷ and the ground truth Y, the downstream evaluation is denied the potential to use X. This exacerbates Evaluation Gap 3 by further abstracting the evaluation statistics from their contexts. For example, X could have been used to disaggregate the evaluation statistics in various dimensions—including for fairness analyses, assuming that socio-demographic data is available and appropriate [6, 9]—or to examine regions of the input space which raise critical safety concerns (e.g., distinguishing a computer vision model’s failure to recognise a pedestrian on the sidewalk from failure to recognise one crossing the road) [3]. Similarly, robustness analyses which compare the model predictions for related inputs in the same neighborhood of the input space are also excluded.

**Assumption 4: Quantifiability.** We have not yet described any modeling assumptions about the mathematical or topological nature of the implied Utility function, which up to now has been conceived as an arbitrary procedure producing an arbitrary output. We observe, however, that when models are evaluated, there is a social desire to produce a small number of scalar scores. This is reinforced by “leaderboadism” [51], and extends to the point of averaging different types of scores such as correlation and accuracy [170]. We identify two assumptions here: first, that impacts on each individual can be reduced to a single numeric value (and thus different dimensions of impacts are commensurable); second,

\[^{3}\text{E.g., one machine learning fairness paper says } \text{“c is the cost of detention in units of crime prevented.” [39]}

![Figure 2: Causal graph illustrating the Input Myopia Assumption.](image)
that impacts across individuals are similarly commensurable. We define \( \hat{y} \in \hat{Y} \) and \( y \in Y \) to be a specific model prediction, and a specific “ground truth” value respectively, leading to the Individual Quantifiability Assumption and the Collective Quantifiability Assumption, respectively.

\[
\text{Utility}(\hat{y}, y) \in \mathbb{R} \quad \text{(Individual Quantifiability Assumption)}
\]

\[
\text{Utility}(\hat{Y}, Y) \approx E(\hat{y}, y)(\text{Utility}(\hat{y}, y)) \quad \text{(Collective Quantifiability Assumption)}
\]

Composing these assumptions with the previous ones leads to the belief that the evaluation can be summarized as a scalar statistic: \( \text{Utility}(\hat{Y}, Y) \in \mathbb{R} \).

**Evaluation Gap 6: Incommensurables.** The Quantifiability Assumptions assume that the impacts on individuals are reducible to numbers, trivializing the frequent difficulty in comparing different benefits and costs [111]. Furthermore, the harms and benefits across individuals are assumed to be comparable in the same scale. These assumptions are likely to disproportionately impact underrepresented groups, for whom model impacts might differ in qualitative ways from the well represented groups [74, 146, 147]. The former groups are less likely to be represented in the ML team [173] and hence less likely to have their standpoints on harms and benefits acknowledged.

**Assumption 5: Failures Cases Are Equivalent.** For classification tasks, common evaluation metrics such as accuracy or error rate model the utility of \( \hat{Y} \) as binary (i.e., either 1 or 0), depending entirely on whether or not it is equal to the “ground truth” \( Y \). That is, for a binary task, \( \text{Utility}(\hat{Y}=0, Y=0)=\text{Utility}(\hat{Y}=1, Y=1) \) and \( \text{Utility}(\hat{Y}=0, Y=1)=\text{Utility}(\hat{Y}=1, Y=0) \). Similarly for regression tasks, common metrics such as MAE and MSE take the magnitude of error into account, yet still treat certain failures as equivalent (specifically, \( \text{Utility}(\hat{Y}=\hat{y}, Y=y+\delta)=\text{Utility}(\hat{Y}=\hat{y}, Y=y-\delta) \), for all \( \delta \).

\[
\text{Utility}(\hat{Y}=\hat{y}, Y=y) \approx 1(\hat{y}=y)
\]

(Asumption of Equivalent Failures [Classification])

\[
\text{Utility}(\hat{y}, y) \quad \text{is a function of } |\hat{y} - y|
\]

(Asumption of Equivalent Failures [Regression])

Taken together with the previous assumptions, this yields \( \text{Utility}(\hat{y}, y) = P(\hat{y} = y) \) for classification tasks.

**Evaluation Gap 7: Disparate harms and benefits.** Treating all failure cases as equivalent fails to appreciate that different classes of errors often have very different impacts [32, 134]. In multiclass classification, severely offensive predictions (e.g., predicting an animal in an image of a person) are given the same weight as inoffensive ones. In regression tasks, insensitivity to either the direction of the difference \( \hat{y} - y \) or the magnitude of \( y \) can result in evaluations being possibly poor proxies for downstream impacts. (One common application use case of regression models is to apply a cutoff threshold \( t \) to the predicted scalar values, for which both the direction of error and the magnitude of \( y \) are relevant.)

**Assumption 6: Test Data Validity.** Taken collectively, the previous assumptions might lead one to use accuracy as an evaluation metric for a classification task. Further assumptions can then be made in deciding how to estimate accuracy. The final assumption we discuss here is that the test data over which accuracy (or other metrics) is calculated provides a good estimate of the accuracy of the model when embedded in the ecosystem.

\[
P(\hat{y} = y) \approx P(\hat{y}' = y')
\]

(Asumption of Test Data Validity [Classification])

where \( y' \) and \( \hat{y}' \) are the ground truth labels and the model predictions on the test data, respectively.

**Evaluation Gap 8: Data Drifts.** A simple model of the ecosystem’s data distributions is particularly risky when system feedback effects would cause the distributions of data in the ecosystem to diverge from those in the evaluation sample [93, 107]. In general, this can lead to overconfidence in the system’s reliability, which can be exacerbated for regions in the tail of the input distribution.

### 4.2 Discussion

We have described six assumptions that simplify the model evaluation task. Taken together, they would cause one to believe—with compounding risks—that a model’s accuracy is a good proxy for its fitness for an application. We sketch this composition of assumptions in Figure 4, along with questions that illustrate the gaps raised by each assumption. Our reason for teasing apart these assumptions and their compounding effects is not to attack the “strawman” of naive application-centric evaluations which rely solely on estimating model accuracy. Rather, our goal is to point out that most model evaluations, even sophisticated ones, make such assumptions to varying degrees. For example:

- Some robustness evaluations (for surveys, see [54, 171]) explicitly tackle the problem of distribution shifts, rejecting the Assumptions of Test Data Validity without questioning the other assumptions we have identified.
- Some sensitivity evaluations consider the effect on the model predictions of small changes in the input, but use accuracy as an evaluation metric, rejecting the Input Myopia Assumption without questioning the others [146].
- Some fairness evaluations perform disaggregated analyses using the Recall or Precision metrics, sticking by all assumptions other than Input Myopia and Equivalent Failures [37, 71].

It may not be possible to avoid all of the assumptions all of the time; nevertheless unavoidable assumptions should be acknowledged and critically examined. The six assumptions we have identified also provide a lens for assessing the consistency of some evaluation metrics with other assumptions that have been made during the evaluation, for example:

- **Is F-score consistent with an utilitarian evaluation framework?** The F-score is mathematically a harmonic mean— which is often appropriate for averaging pairs of rates (e.g., two speeds). When applied to Precision and Recall, however, the F-score constitutes a peculiar averaging of “apples and oranges,” since, when conceived as rates, Precision and Recall measure rates of change of different quantities. [131]. F-score is thus difficult to interpret within an evaluation framework that aims to maximize model utility.
- **Do threshold-free evaluations such as the Area Under the Receiver Operating Characteristic (AUROC) abstract too much of...**
Table 4: Sketch of how the six assumptions of Section 4—when taken collectively—compose to simplify the task of evaluating (unobserved) distributions of ground truth and model predictions, respectively, while the variables $Y$ and $\hat{Y}$ notation (akin to pseudo-code) is used to enable rapid glossing of the main connections. $Y = y$ and $\hat{Y} = \hat{y}$ denote the true (unobserved) distributions of ground truth and model predictions, respectively, while the variables $Y' = y'$ and $\hat{Y}' = \hat{y}'$ denote the samples of reference labels and model predictions over which accuracy is calculated in practice. The order of the assumptions reflects an increasing focus on technical aspects of model evaluation, and a corresponding minimizing of non-technical aspects. Appendix C illustrates how each of the sets of considerations might apply in a hypothetical application of a computer vision model.

| Assumption | Considerations that might be Overlooked |
|------------|----------------------------------------|
| Understand(Model + Ecosystem) | Application-centric evaluation |
| \( \approx \) Utility(Model + Ecosystem) | Consequentialism |
| \( \approx \) Utility($\hat{Y}, X, Y$) | Abstractability from Context |
| \( \approx \) Utility($\hat{Y}, Y$) | Input Myopia |
| \( \approx E(\hat{y}^\prime = \hat{y}, Y = y)\) | Quantitative Modeling |
| \( \approx P(\hat{y} = y)\) | Equivalent Failures |
| \( \approx P(\hat{y}' = y')\) | Test Data Validity |

5 CONTEXTUALIZING APPLICATION-CENTRIC MODEL EVALUATIONS

the deployment context? Since AUROC is calculated by averaging over a range of possible threshold values, it "cannot be interpreted as having any relevance to any particular classifier" [130] (which is not saying AUROC is irrelevant to evaluating the learner, cf. Section 2, nor to a learned model’s propensity to correctly rank positive instances above negative ones). The same argument can be made for the Mean Average Precision metric used in image classification (see Appendix A). For useful application-centric evaluations, it is more meaningful to report pairs of ($\text{Precision}$, $\text{Recall}$) values (for all classes) for a range of threshold values [129].

In both cases, we ask whether such metrics are of limited utility in application-centric evaluations and whether they are better left to learner-centric ones.

5.1 Minding the Gaps between Evaluation Goals and Research Practice

Documenting assumptions made during model evaluation is critical for transparency and enables more informed decisions. If an assumption is difficult to avoid in practice, consider augmenting the evaluation with signals that may shed complementary light on questions of concern. For example, even a handful of insightful comments from members of impacted communities can be an invaluable complement to evaluations using quantitative metrics. We now consider specific mitigation strategies for each of the gaps in turn.

MINDING GAP 1: EVALUATE MORE THAN CONSEQUENCES. To reduce the gap introduced by the Consequentialism Assumption, evaluate the processes that led to the creation of the model, including how datasets were constructed [151]. We echo calls for more reflexivity around social and intentional factors around model development [117], more documentation of the complete lifecycle of model development [83, 168], and greater transparency around ML models and their datasets [13, 62, 118]. It may be appropriate to contemplate whether the model is aligned with the virtues the organization aspires to [166]. Consider the question of whether any ML model could be a morally appropriate solution in this application context, e.g., whether it is appropriate to make decisions about one person on the basis of others’ behaviors [49].
Minding Gap 2: Center Obligations. Since reasoning about uncertain future states of the world is fraught with challenges [29], evaluations should consider indirect consequences and assess how the model upholds social obligations within the ecosystem. This may involve processes such as assessments of human rights, social and ethical impact [110, 115], audits of whether the ML system upholds the organization’s declared values or principles [137], and/or assessments of the potential for privacy leakage (e.g., [30, 176]).

Minding Gap 3: Demarginalize the Context. To address the gap introduced by the Assumption of Abstractability from Context, consider externalities such as energy consumption [75, 153], as well as resource requirements [51]. It is important to think about how the human and technical parts of the system will interact [112, 155]. Note that when substituting one model for another—or for displaced human labor—system stability can itself be a desirable property independent of model accuracy (and perhaps counter to tech industry discourses of “disruption” [64]), and a range of metrics exist for comparing predictions with those of a legacy model [47]. Care should be taken to avoid the “portability trap” of assuming that what is good for one context is good for another [155]. The more attention paid to the specifics of the application context, the better; hence, metrics which assume no particular classification threshold, such as AUC, may provide limited signal for any single context.

Minding Gap 4: Make Subjectivities Transparent. Acknowledge the subjectivities inherent in many tasks [2]. An array of recent scholarship on subjectivity in ML has “embraced disagreement” through practices of explicitly modeling—in both the data model and the ML model—inter-subject variation in interpretations [7, 12, 45, 48, 55]. For the purposes of ML model evaluations, disaggregating labels on test data according to the cultural and socio-demographic standpoints of their annotators enables more nuanced disaggregated evaluation statistics [132].

Minding Gap 5: Respect Differences Between Inputs. A realistic “null hypothesis” is that misclassifications affect people in the application ecosystem disparately. For example, people may differ both in their preferences regarding model predictions \( \hat{Y} \) per se, as well as their preferences regarding model accuracy \( Y = \hat{Y} \) [17]. As such—and independent of fairness considerations—evaluations should be routinely pay attention to different parts of the input distribution, including disaggregating along social subgroups. Special attention should be paid to the tail of the distribution and outliers during evaluation, as these may require further analysis to diagnose the potential for rare but unsafe impacts. Input sensitivity testing can provide useful information about the sensitivity of the classifier to dimensions of input variation known to be of concern (e.g., gender in text [21, 66, 81, 181]).

Minding Gap 6: Think Beyond Scalar Utility. Resist the temptation to reduce a model’s utility to a single scalar value, either for stack ranking [51] or to simplify the cognitive load on decision makers. Instead, include a range of different metrics and evaluation distributions in the evaluation [118]. Acknowledge and report epistemic uncertainty, e.g., the effects of missing data or measurement and sampling error on metrics. Acknowledge qualitative impacts that are not addressed by metrics (e.g., harms to application users caused by supplanting socially meaningful human interactions), and rigorously assess the validity of attempts to measure social or emotional harms. Be conservative in aggregations: consider plotting data rather than reporting summary statistics (cf. Anscombe’s quartet); do not aggregate unlike quantities; report multiple estimates of central tendency and variation; and don’t assume that all users of an application will have equal benefits (or harms) from system outcomes. Consider applying aggregation and partial ranking techniques from the fair division literature to ML models, including techniques that give greater weight to those with the worst outcomes (e.g., in the extreme case, “Maximin”) [50].

Minding Gap 7: Respect Differences Between Failures. If the harms of false positives and false negatives are incommensurable, report them separately. If commensurable, weight each appropriately. For multiclass classifiers, this approach generalizes to a classification cost matrix [164], and, more generally, including the confusion matrix before costs are assigned; for regression tasks, report metrics such as MSE disaggregated by buckets of \( Y \).

Minding Gap 8: Validate Quality of Test Data. For transparency, do not assume it is obvious to others which datasets are used in training and evaluation; instead, be explicit about the provenance, distribution, and known biases of the datasets in use [6]. Consider Bayesian approaches to dealing with uncertainty about data distributions [91, 99, 116], especially when sample sizes are small or prior work has revealed systematic biases. For example, an evaluation which uses limited data in a novel domain (or in an under-studied language) to investigate gender biases in pronoun resolution should be tentative in drawing strong positive conclusions about “fairness” due to abundant evidence of gender biases in English pronoun resolution models (e.g. [172]).

5.2 Alternate Model Evaluation Methodologies
More radical excursions from the disciplinary paradigm are often worth considering, especially in scenarios with high stakes or high uncertainty.

Evaluation Remits. In 1995, Sparck Jones and Galliers called for a careful approach to NLP evaluation that is broadly applicable to ML model evaluations (see Appendix D) [92]. Their approach involves a top-down examination of the context and goal of the evaluation before the evaluation design even begins, and their call for careful documentation of the evaluation “remit”—i.e., official responsibilities—is in line with more recent work calling for stakeholder transparency for ML [83, 137]. They advocate for establishing whose perspectives are adopted in the evaluation and whose interests prompted it. Appendix D sketches how Sparck Jones and Galliers’ framework could be adopted for ML model evaluations.

Active Testing. Active Testing aims to iteratively choose new items that are most informative in addressing the goals of the evaluation [69, 96] (cf. its cousin Active Learning, which selects items that are informative for the learner). Active Testing provides a better estimate of model performance than using the same number of test instances sampled IID. Exploring Active Testing in pursuit

Note that in many real-world applications the “ground truth” variable \( Y \) may be a convenient counterfactual fiction, since the system’s actions on the basis of the prediction \( \hat{Y} \) may inhibit \( Y \) from being realised—for example, a finance ML model may predict a potential customer would default on a loan if given one, and hence the system the model is deployed in may prevent the customer getting a loan in the first place.
of fairness testing goals seems a promising direction for future research.

Adversarial Testing. In many cases, there is great uncertainty regarding an application deployment context. One cautious and conservative approach—especially in the face of great uncertainty—is to simulate “adversaries” trying to provoke harmful outcomes from the system. Borrowing adversarial techniques from security testing and privacy testing, adversarial testing of models requires due diligence to trigger the most harmful model predictions, using either manually chosen or algorithmically generated test instances [52, 145, 177, 179].

Multidimensional Comparisons. When comparing candidate models, avoid the “Leaderboardism Trap” of believing that a total ordering of candidates is possible. A multidimensional and nuanced evaluation may provide at best a partial ordering of candidate models, and it may require careful and accountable judgement and qualitative considerations to decide among them. The Fair Division literature on Social Welfare Orderings may be a promising direction for developing evaluation frameworks that prioritize “egalitarian” considerations, in which greater weighting is given to those who are worst impacted by a model [50].

5.3 Evaluation-driven ML Methodologies

In this section, we follow Rostamzadeh et al. in drawing inspiration from test-driven practices, such as those of software development [144]. Traditional software testing involves significant time, resources, and effort [72]; even moderate-sized software projects spend hundreds of person-hours writing test cases, implementing them, and meticulously documenting the test results. In fact, software testing is sometimes considered an art [121] requiring its own technical and non-technical skills [113, 149], and entire career paths are built around testing [42]. Test-driven development, often associated with agile software engineering frameworks, integrates testing considerations in all parts of the development process [8, 65]. These processes rely on a deep understanding of software requirements and user behavior to anticipate failure modes during deployment and to expand the test suite. (In contrast, ML testing is often relegated to a small portion of the ML development cycle, and predominantly focuses on a static snapshot of data to provide performance guarantees.) These software testing methodologies provide a model for ML testing. First, the model suggests anticipating, planning for, and integrating testing in all stages of the development cycle, research problem ideation, the setting of objectives, and system implementation. Second, build a practice around bringing diverse perspectives into designing the test suite. Additionally, consider participatory approaches (e.g., [112]) to ensure that the test suite accounts for societal contexts and embedded values within which the ML system will be deployed.

An important principle in test-driven software development is visibility into the test data. Typically, engineers working on a system can not only see the details of the test suites but also often develop those test suites themselves. In contrast, the paradigm of ML evaluation methodologies is that the ML practitioner should not inspect the test data, lest their observations result in design decisions that produce an overfitted model. How, then, can these two methodologies be reconciled? We believe that incentives are one important consideration. In the ML research community, the “competition mindset” might indeed lead to “cheating” via deliberate overfitting. In contrast, in real-world applications model developers might benefit from a healthy model ecosystem, for example when they are members of that ecosystem. (However, when developers come from a different society altogether there may be disinterest or disalignment [146].)

Software testing produces artifacts such as execution traces, and test coverage information [72]. Developing practices for routinely sharing testing artifacts with stakeholders provides for more robust scrutiny and diagnosis of harmful error cases [137]. In being flexible enough to adapt to the information needs of stakeholders, software testing artifacts can be considered a form of boundary object [159]. Within an ML context, these considerations point towards adopting ML transparency mechanisms incorporating comprehensive evaluations, such as model cards [118]. The processes that go into building test cases should be documented, so the consumer of the ML system can better understand the system’s reliability. Finally, as for any high-stakes system—software, ML or otherwise—evaluation documentation constitutes an important part of the chain of auditable artifacts required for robust accountability and governance practices [137].

6 CONCLUSIONS

In this paper, we compared the evaluation practices in the ML research community to the ideal information needs of those who use models in real-world applications. The observed disconnect between the two is likely due to differences in motivations and goals, and also pressures to demonstrate “state of the art” performance on shared tasks, metrics and leaderboards [51, 94, 162], as well as a focus on the learner as the object upon which the researcher hopes to shed light. One limitation of our methodology is reliance on published papers, and we encourage more human subjects research in the future, in a similar vein to e.g. [78, 109, 148]. We identified a range of evaluation gaps that risk being overlooked if the ML research community’s evaluation practices are uncritically adopted when for applications, and identify six assumptions that would have to be valid if these gaps are to be overlooked. The assumptions range from a broad focus on consequentialism to technical concerns regarding distributions of evaluation data. By presenting these assumptions as a coherent framework, we provide not just a set of mitigations for each evaluation gap, but also demonstrate the relationships between these mitigations. We show how in the naive case these assumptions chain together, leading to the grossest assumption that calculating model accuracy on data I.I.D. with the training data can be a reliable signal for real-world applications. We contrast the practices of ML model evaluation with those of the mature engineering practices of software testing to draw out lessons for non-I.I.D. testing under a variety of stress conditions and failure severities. One limitation of our analysis is that we are generally domain-agnostic, and we hope to stimulate investigations of assumptions and gaps for specific application domains. We believe it is fundamental that model developers are explicit about methodological assumptions in their evaluations. We believe that ML model evaluations have great potential to enable interpretation and use by different technical and non-technical communities [159].
Evaluation Gaps in Natural Language Understanding? In Proceedings of the the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 4483–4485.

[23] Eric Breck, Jianping Cai, Eric Nielson, Michael Sahlb, and D Sculley. 2017. The ML test score: A rubric for ML production readiness and technical debt reduction. In 2017 IEEE International Conference on Big Data (Big Data). IEEE, 1123–1132.

[24] Leo Breiman. 2001. Statistical modeling: The two cultures (with comments and rejoinder by the author). Statistical science 16, 3 (2001), 199–253.

[25] Benjamin Brewster. 1881. The Yale Literary Magazine October 1881–June 1882 (1881).

[26] William Bullett, Jon Schmidt, Irfan Alvi, Erik Nelson, and Tonatihu Rodriguez-Niik. 2015. Philosophy of engineering: What it is and why it matters. Journal of Professional Issues in Engineering Education and Practice 141, 3 (2015), 02544003.

[27] Ravshan Bunev and Yunfeng Huang. 2010. A utility-driven approach to question ranking in social QA. In Proceedings of The 23rd International Conference on Computational Linguistics (COLING 2010). 125–133.

[28] Ernmo Cai, Da-Cheng Juan, Dimitrios Stamoulis, and Diana Marculescu. 2017. NeuralPower: Predict and deploy energy-efﬁcient convolutional neural networks. In Asian Conference on Machine Learning. PMLR, 622–637.

[29] Dallas Card and Noah A Smith. 2020. On Conceptualism and Fairness. Frontiers in Artificial Intelligence 3 (2020), 34.

[30] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In 36th USENIX Security Symposium (USENIX Security 22). 2663–2680.

[31] Brandon Carter, Siddhartha Jain, Jonas W Mueller, and David Gifford. 2021. Overinterpretation reveals image classiﬁcation model pathologies. Advances in Neural Information Processing Systems 34 (2021).

[32] Robert Challen, Joshua Denny, Martin Pitt, Luke Gompels, Tom Edwards, and Krasimira Tsaneva-Atanasova. 2019. Artiﬁcial intelligence, bias and clinical safety. BMJ Quality & Safety 8, 3 (2019), 253–257.

[33] James I Chilton. 1998. Nothing about us without us. University of California Press.

[34] Mayree Chen, Karan Goel, Nimit S Sollonio, Fait Poms, Kayonovs Fatailohn, and Christopher Re. 2021. Mandoline: Model Evaluation under Distribution Shift in Machine Learning. PMLR, 1617–1629.

[35] Patricia Chilirol, Véronique Moriceau, Farah Benamara, Alda Mari, Gloria Origgi, and Marlene Coulomb-Guly. 2020. He said “who’s gonna take care of your children when you are at ACL?” Reported Sexist Acts are Not Sexist. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 4055–4066.

[36] Alex Cholás-Wood, Madison Coots, Emma Brunskill, and Sharad Goel. 2021. Learning to be Fair: A Conceptualist Approach to Equitable Decision-Making. arXiv preprint arXiv:2109.08792 (2021).

[37] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big data 5, 2 (2017), 153–163.

[38] Sam Corbett-Davies and Sharad Goel. 2018. The measure and mismeasure of fairness: A critical review of fair machine learning. arXiv preprint arXiv:1808.00023 (2018).

[39] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic decision making and the cost of fairness. In Proceedings of the 5th International Conference on Machine Learning. PMLR, 1617–1629.

[40] Eric Breck, Shanqing Cai, Eric Nielson, Michael Sahlb, and D Sculley. 2017. The ML test score: A rubric for ML production readiness and technical debt reduction. In 2017 IEEE International Conference on Big Data (Big Data). IEEE, 1123–1132.

[41] William Bullett, Jon Schmidt, Irfan Alvi, Erik Nelson, and Tonatihu Rodriguez-Niik. 2015. Philosophy of engineering: What it is and why it matters. Journal of Professional Issues in Engineering Education and Practice 141, 3 (2015), 02544003.

[42] Ravshan Bunev and Yunfeng Huang. 2010. A utility-driven approach to question ranking in social QA. In Proceedings of The 23rd International Conference on Computational Linguistics (COLING 2010). 125–133.

By naming each assumption we identify and exploring its technical and sociological consequences, we hope to encourage more robust interdisciplinary debate and, ultimately, to nudge model evaluation practice away from abundant opaque unknowns.

ACKNOWLEDGMENTS
We acknowledge useful feedback from Daniel J. Barrett, Alexander D’Amour, Stephen Pfahl, D. Sculley, and the anonymous reviewers.

REFERENCES
[1] Firoj Alam, Shafiq Joty, and Muhammad Imran. 2018. Domain Adaptation with Adversarial Training and Graph Embeddings. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 1077–1087.

[2] Cecilia Ovenseder Alm. 2011. Subjective natural language problems: Motivations, applications, characterizations, and implications. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 107–112.

[3] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in AI safety. arXiv preprint arXiv:1606.06565 (2016).

[4] Adam J Andreetta, Ninn Kirkham, and Marco Rizzi. 2021. AI, big data, and the four forms of consent. In E & T Society, 2021. 1–14.

[5] McKane Andrus and Thomas K Gilbert. 2019. Towards a just theory of measurement: A principled social measurement assurance program for machine learning. In Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society. 445–451.

[6] McKane Andrus, Elena Spitzer, Jeffrey Brown, and Alice Xiang. 2021. What We Can’t Measure, We Can’t Understand: Challenges to Demographic Data Procurement in the Pursuit of Fairness. In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. 249–260.

[7] Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven ways of seeing. AI Magazine 36, 1 (2015), 15–24.

[8] Dave Astels. 2005. Test driven development: A practical guide. Prentice Hall Professional Technical Reference.

[9] Solon Barocas, Anshong Guo, Ece Kamar, Iacqueyln Krones, Meredith Ringel Morris, Jennifer Wortman Vaughan, Duncan Waddow, and Hanna Wallach. 2021. Designing Disaggregated Evaluations of AI Systems: Choices, Considerations, and Tradeoffs. arXiv preprint arXiv:2103.06676 (2021).

[10] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2017. Fairness in machine learning. NIPS tutorial 1 (2017), 2017.

[11] Roland Barthes. 1977. Image-Music-Text. Macmillan.

[12] Valerio Basile, Federico Cabieta, Andrea Campagnoli, and Michael Fell. 2021. Toward a Perspectivist Turn in Ground Truthing for Predictive Computing. arXiv preprint arXiv:2109.04270 (2021).

[13] Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science.Transactions of the Association for Computational Linguistics 6 (2018), 587–604.

[14] Emily M Bender, Timnit Gebru, Angelina McNamill-Mayor, and Shmargaret Shumitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. 610–623.

[15] Yoshua Bengio, Yann Lecun, and Geoffrey Hinton. 2012. Deep learning. Artificial Intelligence 294, 7 (2021), 58–65.

[16] John Berger. 2008. Ways of seeing. Penguin UK.

[17] Reuben Binns. 2018. Fairness in machine learning: Lessons from political philosophy. In Conference on Fairness, Accountability and Transparency: PMLR, 149–159.

[18] Abeba Birhane, Pratyusha Kalluri, Dallas Card, William Agnew, Ravit Dotan, and Michelle Bao. 2021. The values encoded in machine learning research. arXiv preprint arXiv:2106.15380 (2021).

[19] Damiañ Blasi, Antonios Anastasopoulos, and Graham Neubig. 2021. Systematic Inequalities in Language Technology Performance across the World’s Languages. arXiv preprint arXiv:2110.06733 (2021).

[20] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021).

[21] Daniel Borkan, Lucas Dixon, Jeffrey Sorenson, Nithum Thain, and Lucy Vasserman. 2019. Measuring metrics for measuring unintended bias with real-data for text classification. In Companion proceedings of the 2019 world wide web conference. 491–500.

[22] Samuel Bowman and George Dahl. 2021. What Will it Take to Fix Benchmarking in Natural Language Understanding? In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 4483–4485.
### APPENDIX A: METRICS IN ML MODEL EVALUATIONS

Here we give definitions and categorizations of some of the metrics reported in the study in Section 3. In practice, there was a long tail since many metrics were used in only a single paper. Here we include only the metrics which were most frequently observed in our study.

| Metric          | Example Task(s)   | Metric category | Definition                                                                 |
|-----------------|-------------------|-----------------|-----------------------------------------------------------------------------|
| Accuracy        | Classification    | ACCURACY        | A metric that penalizes system predictions that do not agree with the reference data \( \frac{TP+TN}{TP+TN+FP+FN} \). |
| AUC             | Classification    | AUC             | The area under the curve parameterized by classification threshold \( t \), typically with \( y \)-axis representing recall and \( x \)-axis representing false positive rate \( \frac{FP}{TP+TN} \). |
| BLEU            | Machine translation | PRECISION       | A form of "\( n \)-gram precision," originally designed for machine translation but also sometimes used for other text generation tasks, which measures whether sequences of words in the system output are also present in the reference texts [126]. |
| Dice            | Image segmentation | OVERLAP         | Equivalent to \( F_1 \) \( \frac{2TP}{2TP+FP+FN} \). More commonly used for medical image segmentation. |
| Error rate      | Classification    | ACCURACY        | The inverse of accuracy \( \frac{TP+TN}{TP+TN+FP+FN} \). |
| \( F \) (or \( F_1 \)) | Text classification | OVERLAP         | The harmonic mean of recall and precision \( \frac{PR}{P+R} \), originally developed for information retrieval [167] but now widely used in NLP. |
| \( F_{0.5} \)   | Text classification | OVERLAP         | A weighted harmonic mean of recall and precision, with greater weight given to recall \((1 + \beta^2) \frac{PR}{P+R}\) with \( \beta = 0.5 \). |
| Hausdorff distance | Medical Image Segmentation | DISTANCE       | A measure of distance between two sets in a metric space. Two sets have a low Hausdorff distance if every point in each set is close to a point in the other set. |
| IoU             | Image segmentation | OVERLAP         | Equivalent to Jaccard. Has been argued to address shortcomings in \( F_1 \)'s asymmetry with respect to classes \( \frac{TP+TN}{TP+TN+FP+FN} \). In information retrieval, the average over information needs of the average precision of the documents retrieved for that need. The area under the Precision-Recall tradeoff curve, averaged over multiple IoU (intersection over union) threshold values, then averaged across all categories (https://cocodataset.org/#detection-eval). |
| Matthew’s Correlation Coefficient | Image segmentation | CORRELATION     | A measure of ranking quality which takes into account the usefulness of items based on their ranking in the result list. |
| Mean absolute error | Regression     | DISTANCE        | In information retrieval, the average over information needs of the average precision of the documents retrieved for that need. |
| Mean Average Precision (MAP) | Information retrieval (NLP) | AUC             | A measure for evaluating processes that produces an ordered list of possible responses. The average of the inverse rank of the first relevant item retrieved. |
| Mean average precision (mAP) | Object detection (CV) | AUC             | Mean squared error (MSE) measures the average of the squared difference between estimated and actual values. |
| Mean reciprocal rank | Information retrieval | OTHER           | A measure of ranking quality which takes into account the usefulness of items based on their ranking in the result list. |
| MSE             | Image Decomposition | DISTANCE        | A measure for evaluating processes that produces an ordered list of possible responses. The average of the inverse rank of the first relevant item retrieved. |

Normalized Discounted Cumulative Gain (NDCG)
| Metric          | Example Task(s)         | Metric category | Definition                                                                 |
|----------------|-------------------------|-----------------|-----------------------------------------------------------------------------|
| Pearson’s $r$  | Quality Estimation      | CORRELATION     | A measure of linear correlation between two sets of data.                   |
| Perplexity     | Language modeling       | PERPLEXITY      | Information-theoretic metric (measured in bits-per-unit, e.g., bits-per-character or bits-per-sentence) often used for language models, inversely related to the probability assigned to the test data by the model. Closely related to the cross-entropy between the model and the test data. Can be thought of as how efficiently does the language model encode the test data. |
| Precision      | Classification          | PRECISION       | A metric that penalizes the system for predicting a class (if class is unspecified, by default the “positive” class) when the reference data did not belong to this class ($\frac{TP}{TP+FP}$). |
| PSNR           | Super Resolution        | DISTANCE        | Peak Signal-to-Noise ratio (PSNR) is the ratio between the maximum possible value of a signal and the power of distorting noise (Mean Squared Error) that impacts the quality of its representation. |
| Recall         | Classification          | RECALL          | Also known as “sensitivity”, this metric that penalizes the system for failing to predict a class (if class is unspecified, by default the “positive” class) when the reference data did belong to this class ($\frac{TP}{TP+FN}$); a.k.a. true positive rate. |
| RMSE           | Depth Estimation        | DISTANCE        | Root Mean Square Error (RMSE) is the square root of the MSE.                |
| Rouge          | Text summarization      | RECALL          | A form of “n-gram recall,” originally designed for text summarization but also sometimes used for other text generation tasks, which measures whether sequences of words in the reference texts are also present in the system output. |
| Spearman’s $\rho$ | Graph Edit Distance    | CORRELATION     | A measure of monotonic association between two variables–less restrictive than linear correlations. |
| Specificity    | Classification          | OTHER           | Like Precision, this metric that penalizes the system for failing to predict a class (if class is unspecified, by default the "positive" class) when the reference data did belong to this class; unlike Precision it rewards true negatives rather than true positives ($\frac{TN}{TN+FN}$). |
| SSIM           | Super Resolution        | DISTANCE        | The Structural Similarity Method (SSIM) is a perception-based method for measuring the similarity between two images. The formula is based on comparison measurements of luminance, contrast, and structure. |
| Top-\(n\) accuracy | Face recognition      | ACCURACY         | A metric for systems that return ranked lists, which calculates accuracy over the top \(n\) entries in each list. |
| Word error rate | Speech recognition      | ACCURACY         | The inverse of word accuracy: \(1 - \text{word accuracy}\) (which is not technically always in [0, 1] due to the way word accuracy is defined but which is categorized as “Accuracy” here because both insertions and deletions are penalized). |

Table 5: Definitions and categorizations of metrics reported in Section 3. TP, TN, FP and FN indicate the number of true positives, true negatives, false positives and false negatives, respectively. \(y\) and \(\hat{y}\) represent actual values and values predicted by the system, respectively.
APPENDIX B: TYPES OF EVALUATION DATA USED IN ML MODEL EVALUATIONS

| Type of Test Data | Example Task(s)       | I.I.D. with training data? | Definition |
|-------------------|-----------------------|----------------------------|------------|
| Test split        | Classification        | yes                        | Typically, labeled data is partitioned into training and test splits (and often a dev split too), drawn randomly from the same dataset. |
| Manual resource   | Lexical acquisition   | no                         | A manually compiled resource (in NLP, often a word-based resource such as a lexicon or thesaurus), against which knowledge acquired from a dataset is compared. |
| References        | Machine translation   | no                         | Reference outputs (typically obtained prior to building the system) which a generative system is trying to reproduce, typically obtained from humans (e.g., manual translations of input sentences in the case of evaluations using BLEU for machine translation tasks). |
| Training data     | Keyword extraction    | yes                        | Training data that contains labels is used to evaluate an unsupervised algorithm that did not have access to the labels during learning. |
| Novel distribution| Domain transfer       | no                         | Test data that has the same form as the training data but is drawn from a different distribution (e.g., in the case of NLP training on labeled newspaper data and testing on labeled Wikipedia data). |

Table 6: Types of datasets used in ML model evaluations.

APPENDIX C: EXAMPLE OF ASSUMPTIONS AND GAPS FOR A HYPOTHETICAL APPLICATION

Suppose we are evaluating a hypothetical image classification model for use in an application for assisting blind people in identifying groceries in their pantries. Then some application-specific questions related to the assumptions in Section 4 might be:

**Consequentialism.** Was data ethically sourced and labeled? Were blind people involved in the design process? Does this use of this model encourage high-risk uses of other similar models, such as identifying pharmaceutical products?

**Abstractability from Context.** Does the application have a human-in-the-loop feature available when the model is uncertain? Will the system nudge purchasing behaviors towards products on which the model performs well?

**Input Myopia.** Are uncommon grocery products misclassified more often? Does this disproportionately impact home cooks who don’t stick to the dominant cuisines, or who have food requirements due to medical conditions?

**Quantitative Modeling.** Does measuring predictive accuracy fail to take into account dignitary consequences associated with increased independence? Should each user be weighted equally in the evaluation (cf. each image)?

**Equivalent Failures.** Are there severe risks in confusing certain pairs of products, e.g., food products associated with dangerous allergies? Are some errors only minimally inconvenient, such as confusing different shapes of pasta?

**Test Data Validity.** Is the evaluation data representative of what the application’s users have in their pantries? Are the image qualities (lighting, focus, framing, etc.) representative of images taken by blind photographers?
APPENDIX D: MODEL EVALUATION REMITS AND DESIGN

Model Evaluation Remit

To establish:
- motivation — why evaluate the model?
  what is the perspective being adopted — task/financial/administrative/scientific/...
  whose interests prompted the evaluation — developer/funder/...
  who are the consumers of the model evaluation results — manager/user/researcher/...
- goal — what do we want/need to discover?
  orientation — intrinsic/extrinsic
  kind — investigation/experiment
  type — black box/glass box
  form (of yardstick) — ideal/attainable/exemplar/given/judged
- style — suggestive/indicative/exhaustive
- mode — quantitative/qualitative/hybrid

Model Evaluation Design

To identify:
- ends — what is the model for? what is its objective or function?
- context — what is the ecosystem the model is in? what are the animate and inanimate factors?
- constitution — what is the structure of the model? what was the training data?

To determine:
- factors that will be tested
  environment variables
  ‘system’ parameters
- evaluation criteria
- metrics/measures
- methods
- Evaluation data — what type, status and nature?
- Evaluation procedure

Table 7: A sketch of how Karen Sparck Jones and Julia Galliers’ 1995 NLP evaluation framework questionnaire [92] can be adapted for the evaluation of ML models. The output of the remit and the design is a strategy for conducting the model evaluation. For a related but simpler framework based on model requirements analysis, see also the "7-step Recipe" for NLP system evaluation (https://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html) developed by the EAGLES Evaluation Working Group in 1999, which considers whether different parties have a shared understanding of the evaluation’s purpose.