Bulletin of the Technical Community on

Data Engineering

March 2024 Vol. 48 No. 1

IEEE Computer Society

Letters

Special Issue on Data-centric Responsible AI

Conference and Journal Notices

Editorial Board

Editor-in-Chief

Haixun Wang
Instacart
50 Beale Street
San Francisco, CA, 94107
haixun.wang@instacart.com

Associate Editors

Steven Euijong Whang
KAIST
Korea

Karthik Subbian
Amazon
Palo Alto, California, USA

Themis Palpanas
University of Paris
Paris, France

Xin Luna Dong, Alon Halevy
Meta (Facebook)
Menlo Park, California, USA

Distribution

Brookes Little
IEEE Computer Society
10662 Los Vaqueros Circle
Los Alamitos, CA 90720
eblittle@computer.org

The TC on Data Engineering

Membership in the TC on Data Engineering is open to all current members of the IEEE Computer Society who are interested in database systems. The TCDE web page is http://tab.computer.org/tcde/index.html.

The Data Engineering Bulletin

The Bulletin of the Technical Community on Data Engineering is published quarterly and is distributed to all TC members. Its scope includes the design, implementation, modelling, theory and application of database systems and their technology.

Letters, conference information, and news should be sent to the Editor-in-Chief. Papers for each issue are solicited by and should be sent to the Associate Editor responsible for the issue.

Opinions expressed in contributions are those of the authors and do not necessarily reflect the positions of the TC on Data Engineering, the IEEE Computer Society, or the authors' organizations.

The Data Engineering Bulletin web site is at http://tab.computer.org/tcde/bull_about.html.

TCDE Executive Committee

Chair

Murat Kantarcioglu
University of Texas at Dallas

Executive Vice-Chair

Karl Aberer
EPFL

Executive Vice-Chair

Thomas Risse
Goethe University Frankfurt

Vice Chair

Erich J. Neuhold
University of Vienna, Austria

Vice Chair

Malu Castellanos
Teradata Aster

Vice Chair

Xiaofang Zhou
The University of Queensland

Editor-in-Chief of Data Engineering Bulletin

Haixun Wang
Instacart

Diversity & Inclusion and Awards Program Coordinator

Amr El Abbadi
University of California, Santa Barbara

Chair Awards Committee

S Sudarshan
IIT Bombay, India

Membership Promotion

Guoliang Li
Tsinghua University

TCDE Archives

Wookey Lee
INHA University

Advisor

Masaru Kitsuregawa
The University of Tokyo

SIGMOD Liaison

Fatma Ozcan
Google, USA

Letter from the Editor-in-Chief

The pervasive integration of AI into daily life accentuates the imperative of advancing Data-centric Responsible AI. This endeavor transcends mere enhancements in model performance, aiming to ensure that models are not only efficient but also trustworthy, fair, robust, private, secure, and interpretable. Central to this mission is the focus on refining the underlying data, acknowledging the integral role of data quality in determining AI's efficacy.

In this March edition of the Data Engineering Bulletin, we delve into data-driven approaches and the ethos of responsible AI. This issue, meticulously curated by Steven Euijong Whang, showcases five papers that chart a course toward a more equitable, reliable, and secure AI landscape. These contributions not only spotlight prevailing challenges but also introduce pioneering solutions and frameworks poised to foster a more just AI ecosystem.

Specifically, the featured papers elucidate a spectrum of strategies to bolster AI accountability across the machine learning pipeline. This includes innovating data coverage techniques to rectify underrepresentation of minority groups, applying causal modeling to amend biases and enhance data integrity, and devising algorithmic approaches to improve fairness and robustness in aggregating preferences. Moreover, investigations into the resilience of Large Language Models, such as ChatGPT, against adversarial threats, alongside comparisons between food safety practices and data traceability, provide fresh insights on promoting AI's reliability and accountability.

As we explore these scholarly contributions, we are collectively reminded of the shared duty among researchers, practitioners, and policymakers to guide AI's trajectory toward outcomes that are ethically sound and universally beneficial.

Haixun Wang
Instacart

Letter from the Special Issue Editor

Data-centric Responsible AI is becoming increasingly critical as AI is widely used in our everyday lives. In addition to simply improving model performance, it is important to make sure the trained model is trustworthy and responsible in the sense that it is fair, robust, private, secure, explainable, aligns with values, and more. Moreover, AI is only as good as its data, so we must take a data-centric approach and improve the data itself to fundamentally solve these problems. Recently applications like Large Language Models (LLMs) have remarkable performance largely because of the large amounts of data they are trained on, so data-centric research is only going to become more important in the future. This issue is thus timely and contains recent solutions by leading experts in this field.

The first three papers propose Data-centric Responsible AI methods that can be applied at different stages in the machine learning pipeline. The paper *Coverage-based Data-centric Approaches for Responsible and Trustworthy AI* by Shahbazi et al. proposes data coverage methods to identify and resolve misrepresentation of minorities in data. The goal is to identify and resolve insufficient data coverage and generate data-centric reliability warnings to help data scientists determine if a prediction is reliable. Recent generative AI and foundation models can benefit from these techniques to effectively augment datasets with synthetic data. Next, the paper *Overcoming Data Biases: Towards Enhanced Accuracy and Reliability in Machine Learning* by Zhu and Salimi explores how causal modeling can improve the data cleaning, preparation, and quality management for machine learning. Causal reasoning can effectively identify and correct data biases resulting from missing data, confounding variables, and measurement errors and thus improve the fairness and accuracy of machine learning models. Finally, the paper *Fairness and Robustness in Answering Preference Queries* by Roy outlines algorithmic challenges and directions for systematically changing the original aggregated output to satisfy different criteria related to fairness and robustness. The author considers different scenarios on how users provide their input preferences and how the individual preferences get aggregated.

The next two papers explore interesting domains that provide inspiration to further advance Data-centric Responsible AI. The paper *On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective* by Wang et al. performs a thorough evaluation of the robustness of ChatGPT and other LLMs from an adversarial and out-of-distribution perspective. While LLMs are receiving significant attention nowadays, their robustness to unexpected inputs is still understudied, which is a concern especially for safety-critical applications. The authors leverage multiple recent datasets for adversarial robustness and show that ChatGPT performs better and others, but also has much room for improvement. The paper *Red Onions, Soft Cheese and Data: From Food Safety to Data Traceability for Responsible AI* by Grafberger et al. makes the interesting analogy that data traceability for Responsible AI is akin to ensuring food safety. In particular, the U.S. Food and Drug Administration (FDA) detects outbreaks of foodborne illnesses, discovers contaminated food, and conducts traceback investigations through the food supply chain to determine the root cause and issue a comprehensive product recall. Taking inspiration from this process, the authors propose a data-centric vision for Responsible AI that involves prediction monitoring, data tracing, and identifying contaminated data and pipeline steps through audits.

Overall, these works represent the state-of-the-art data management approaches for Data-centric Responsible AI from various angles. We are just scratching the surface, and the data management community is well positioned to eventually realize this vision.

Steven Euijong Whang
Korea Advanced Institute of Science and Technology

Coverage-based Data-centric Approaches for Responsible and Trustworthy AI*

Nima Shahbazi
University of Illinois Chicago
nshahb3@uic.edu

Mahdi Erfanian
University of Illinois Chicago
merfan2@uic.edu

Abolfazl Asudeh
University of Illinois Chicago
asudeh@uic.edu

Abstract

The grand goal of data-driven decision systems is to help make decisions easier, more accurate, at a higher scale, and also just. However, data-driven algorithms are only as good as the data they work with. Yet, data sets, especially those with social data, often do not represent minorities. The paucity of training data is a perpetual problem for AI, and the outcome of ML models for cases not represented in their training data is often not reliable. Hence, without properly addressing the lack of representation issues in data, we cannot expect AI-based societal solutions to have responsible and trustworthy outcomes.

This paper focuses on data coverage as a data-centric approach for identifying and resolving misrepresentation of minorities in data. To achieve this goal, we propose novel algorithms that (a) identify and resolve insufficient data coverage across data with different modalities and (b) use lack of representation information to generate data-centric reliability warnings.

1 Introduction

Data-driven decision-making has shaped every corner of human life, spanning from autonomous vehicles to healthcare and even predictive policing and criminal justice. A pivotal concern, especially in applications that affect individuals, revolves around the reliability of the decisions rendered by the system. It is easy to see that the accuracy of a data-driven decision depends, first and foremost, on the data used to make it. Essentially, the system learns the phenomena that data represent. While we may desire that the data should represent the underlying data distribution from which the production data is drawn, this alone may be insufficient, as it merely enables the model to perform well for the average case. As a result, a model with a high accuracy could fail for specific regions in the data with insufficient representation. These regions may matter because they frequently represent some minority population in society. They could also represent cases that may not happen very often but have a relevant impact on the correctness of a critical decision. In short, if the data fails to sufficiently represent a specific population, the outcome of the decision system for that population may not be trustworthy.

The phenomenon known as *Representation Bias* can arise from how the data was originally collected, or it could be the result of biases introduced post-collection—whether historically, cognitively, or statistically.

Copyright 2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

*This research was supported by the National Science Foundation under grant No. 2107290.

Representation bias is essentially inevitable without a systematic approach to data collection. For example, in the context of survey data collection, vital steps involve identifying all populations within the underlying distribution based on desired demographic information and ensuring comprehensive coverage with sufficient samples from each group. Even then, only an (uncontrolled) subset of the invitees will opt-in to respond to the survey. Another challenge lies in the fact that data scientists often lack control over the data collection process, leading to the reliance on "found data" in the majority of data-driven systems. Therefore, with no guarantee on the aforementioned steps in the data collection process, the found data is most likely a biased sample. Acknowledging the potential harms of representation bias, the notion of *Data Coverage* [1, 2] has been proposed to ensure the adequate representation of minority groups in data sets employed for decision-making and developing sophisticated data science tools.

Addressing representation issues in data poses various challenges depending on the modality of the data. In this paper, we focus on identifying and resolving lack of coverage issues in data with different modalities. We start by proposing a variety of techniques (spanning from geometric and combinatorial optimization to crowd-souring) aimed at efficiently detecting insufficient coverage on structured data sets with non-ordinal categorical and continuous attributes, as well as image data sets. Next, we propose a range of approaches grounded in data integration and generative data augmentation to address the lack of coverage by enriching the data sets with more data. However, with limited control over the data collection processes, it could be difficult and expensive to resolve all misrepresentations. Since adding more data is not always possible, we proceed to introduce data-centric preventive solutions that warn the user about the reliability of their predictions regarding representation bias issues. These warnings assist users in determining whether they trust the outcomes of the models or exercise caution.

2 Detecting Insufficient Representation of Minorities

Representation bias happens when the development (training data) population under-represents and subsequently fails to generalize well for some parts of the target population, due to historical bias, sampling bias, etc. The notion of *data coverage* has been studied across different settings in [2] as a metric to measure representation bias. At a high level, coverage is referred to as having enough similar entries for each object in a data set. For a better understanding, let us go over the definition of the generalized notion of coverage:

Definition 2.1 (Data Coverage) *Consider a data set $\mathcal{D}$ with n tuples, each consisting of d attributes of interest $\mathbf{x} = \{x_1, x_2, \cdots, x_d\}$, such as* `gender`*,* `race`*,* `salary`*,* `age`*, etc, that are used for coverage identification. The data set also contains target attributes $\mathbf{y} = \{y_1, \cdots, y_{d'}\}$ that may or may not be considered for the coverage problem. A query point q is not covered by the data set $\mathcal{D}$, if there are not "enough" data points in $\mathcal{D}$ that are representative of q. To generalize the notion of coverage, let us define $\mathbf{g}(q)$ as the universe of tuples that would represent q and let $\mathbf{g}_\mathcal{D}(q) = \mathbf{g}(q) \cap \mathcal{D}$. In other words, $\mathbf{g}_\mathcal{D}(q)$ are the set of tuples in $\mathcal{D}$ that represent q. Using this notation, we define the coverage of q as the size of $\mathbf{g}_\mathcal{D}(q)$. That is, $cov(q, \mathcal{D}) = |\mathbf{g}_\mathcal{D}(q)|$. Given a value τ, q is covered if $cov(q, \mathcal{D}) > \tau$. Similarly, a group $\mathbf{g}$ is not covered if $\mathbf{g} \cap \mathcal{D} < \tau$. The uncovered region in a data set is the collection of groups that are not covered by it.*

2.1 Structured Data

In this section, we focus on identifying representation bias in structured data. Depending on the type of the attributes of interest, we categorize the techniques into two classes based on whether they target the problem for non-ordinal *categorical* (e.g. `race`, `gender`) or ordinal *continuous* (e.g. `age`) attributes. The attributes of interest considered for representation bias often include sensitive attributes such as `race` and `gender` but are not necessarily limited to them.

2.1.1 Categorical Attributes

For cases where attributes of interest are non-ordinal categorical, the cartesian product of values on a subset of attributes $\mathbf{x}' \subseteq \mathbf{x}$, form a set of (sub-)groups. For example, $\{$ `white male`, `white female`, `black male` $, \cdots \}$ are the subgroups defined on the attributes `(race,gender)`. We refer to the number of attributes used to specify a subgroup as the *level* of that subgroup. For example, the level of the subgroup `white male` is 2, while the level of the subgroup `male` is 1. We use $\ell(\mathbf{g})$, to refer to the level of a subgroup $\mathbf{g}$. Similarly, we say a subgroup $\mathbf{g}'$ is a subset of $\mathbf{g}$, if the groups specifying $\mathbf{g}'$ are a superset of the ones for $\mathbf{g}$. For example `(married white male)` a subset of the more general group `(white male)`. That is, the set of individuals in group `(married white male)` are a subset of `(white male)`. Moreover, we say a subgroup $\mathbf{g}$ is a *parent* of the subgroup $\mathbf{g}'$, if $\mathbf{g}' \subset \mathbf{g}$ and $\ell(\mathbf{g}) = \ell(\mathbf{g}') + 1$. For example, the subgroup `(white male)` is a parent of the subgroup `(married white male)`. We use *patterns* to refer to uncovered subgroups. A pattern P is a string of d values, where $P[i]$ is either a value from the domain of x_i, or it is "unspecified", specified with X. For example, consider a data set with three binary attributes of interest $\mathbf{x} = \{x_1, x_2, x_3\}$. The pattern $P = X01$ specifies all the tuples for which $x_2 = 0$ and $x_3 = 1$ (x_1 can have any value). The set of patterns that identify most general uncovered subgroups are called *Maximal Uncovered Patterns* (MUPs).

No polynomial time algorithm can guarantee the enumeration of the entire MUPs, however, several algorithms inspired by set enumeration and the Apriori algorithm for association rule mining are proposed to efficiently address this problem [1]. In this regard, we introduce *Pattern Graph* data structure that exploits the relationship between patterns to do less work than computing all uncovered patterns by removing the non-maximal ones. The parent-child relationship between the patterns is represented in a graph that can be used to find better algorithms. *Pattern-Breaker* starts from the top of the graph where the general patterns are and moves down by breaking each pattern into more specific ones. If a pattern is uncovered, then all of its descendants are also uncovered and they can not be an MUP, even if they have a parent that is covered. Therefore, this subgraph of the pattern graph can be pruned. The issue with *Pattern-Breaker* is that it explores the covered regions of the pattern graph and for the cases where there are a few uncovered patterns, it has to explore a large portion of the exponential-size graph. To tackle this, *Pattern-Combiner* algorithm is proposed that performs a bottom-up traversal of the pattern graph. It uses an observation that the coverage of a node at the level of the pattern graph can be computed as the sum of the coverage values of its children. The problem with *Pattern-Combiner* is that it traverses over the uncovered nodes first and therefore, it will not perform well for the cases in which most of the nodes in the graph are uncovered. In fact, for the cases where most of the MUPs are placed in the middle of the graph, both *Pattern-Breaker* and *Pattern-Combiner* will not be as efficient as they should traverse half of the graph. Therefore, we propose *Deep-Diver*, a search algorithm based on Depth-First-Search that quickly finds the MUPs, and uses them to limit the search space by pruning the nodes both dominating and dominated by the discovered MUPs.

2.1.2 Continuous Attributes

Data in the real world often consists of a combination of continuous and discrete values. While simple solutions like binning `age` into `young` and `old` can transform the continuous space into discrete. However, they may lead to coarse groupings that are sensitive to the thresholds chosen. It may be inappropriate to treat a 35-yo as `young` but a 36-yo as `old`. Therefore, we extend the notion of coverage to continuous space. Particularly, given data set $\mathcal{D}$ with n tuples over d attributes, and vicinity radius ρ and coverage threshold k, we want to identify the uncovered region – the universe of uncovered query points. A query point in continuous data space is covered if there are enough (at least k) data points in its ρ-vicinity neighborhood. ρ-vicinity neighborhood is the circle centered at the query point with radius ρ.

Depending on the number of attributes in a data set, we propose two algorithms for identifying uncovered regions in data [3]. The first algorithm known as *Uncovered-2D* studies coverage over two-dimensional data sets where $\mathbf{x} = \{x_1, x_2\}$. To find the number of circles that a query point falls into and consequently discover

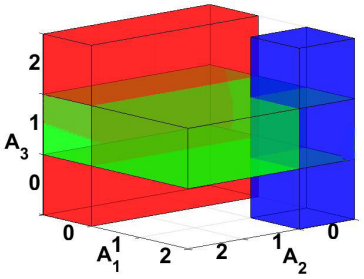


Figure 1: Categorical attributes: the uncovered region of a toy example, as the collection of three MUPs.

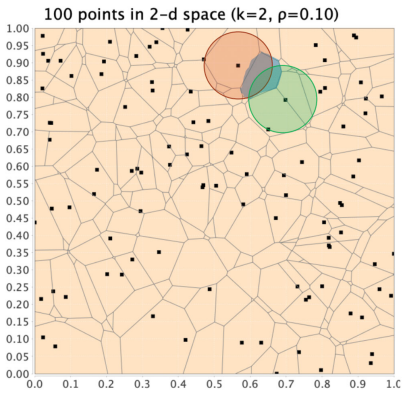


Figure 2: Continuous attributes, 2D: identifying the covered region in the gray Voronoi cell.

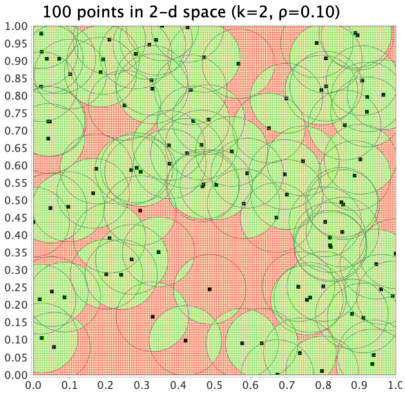


Figure 3: Continuous attributes, 2D: Uncovered region marked in red.

the uncovered region, *Uncovered-2D* makes a connection to k-th order Voronoi diagrams. Consider a data set $\mathcal{D}$ and its corresponding k-th order Voronoi diagram. For every tuple $t \in \mathcal{D}$, let $\circ_t$ be the d-dimensional sphere (d-sphere) with radius ρ centered at t. Consider a k-voronoi cell $\mathcal{V}(S)$ in the k-th order Voronoi diagram $V_k(\mathcal{D})$. Any point q inside the intersections of the d-spheres of tuples in S, i.e. $q \in \bigcap_{\forall t \in S} \circ_t$, is covered, while all other points in the region are uncovered. The algorithm starts by constructing the k-th order Voronoi diagram of the data set and then for each Voronoi cell $\mathcal{V}(S)$ in the diagram, it computes the intersection of the circles of the tuples in S and marks the portion of $\mathcal{V}(S)$ that falls outside it as uncovered. After identifying the uncovered region, a 2D map of $\{x_1, x_2\}$ value combinations is used to report the region to the user. The algorithm for the 2D case can be extended to the general case by relaxing the assumption on the number of attributes to discover the exact uncovered region, however, due to the curse of dimensionality, the search size space explodes as the number of dimensions increases and as a result, the algorithm will not be practical. Therefore, we propose a randomized approximation algorithm based on the geometric notion of ε-**net**. Let $\mathcal{X}$ be a set and $\mathcal{R}$ be a set of subsets of $\mathcal{X}$. A set $\mathcal{N} \subset \mathcal{X}$ is an ε-**net** for $\mathcal{X}$ if for any range $r \in \mathcal{R}$, if $|r \cap \chi| > \varepsilon|\chi|$, then r contains at least one point of N. The idea, at a high level, is to draw enough random samples from the space of potential query points to form an ε-**net**. We then label the sampled query points as $\{-1, +1\}$ depending on whether those are covered or not, and learn the uncovered regions using the samples.

2.2 Image Data

Many known incidents of machine failures due to the lack of representation were on image data. We consider an image data set with a fixed number of low-cardinality sensitive attributes such as `race` and `gender`. It is common that image data sets *lack explicit values* for sensitive attributes, which are crucial for coverage identification. An image data set is often a collection of images from different domains with little to no information about their domain and which groups they belong to. As a result, even studying coverage over low-cardinality and categorical attributes of interests is challenging in these cases.

In Figure 4, we show that due to the issues such *machine bias* and *lack of distribution generalizability*, solely relying on state-of-the-art machine learning (ML) techniques fail to effectively identify lack of coverage in image data sets. Therefore, we propose an approach based on combining crowdsouring with ML [4]. Crowdsourcing is particularly promising for image data, for tasks such as image labeling, which, while challenging for the machine, are "easy" for human beings to conduct with minimal error.

A key observation that enables a cost-effective crowdsourcing approach is that, while studying coverage, we would only like to find out if there are *enough tuples from each subgroup*. Suppose a subgroup is covered if there are $\tau = 100$ instances of it in the data set. Assume the (majority) group $\mathbf{g}_1$ contains $n_1 \gg 100$ objects in the data

set. To verify that $\mathbf{g}_1$ is covered, it is enough for the crowd to discover 100 of those objects, not the entire n_1. Following this, $O(\tau)$ provides a lower bound on the number of crowd tasks required to verify a given group is covered. Still, this lower bound only holds for the groups that are covered, i.e., there is at least τ of those in the data set. Surprisingly, verifying that a minority group is indeed uncovered is cumbersome, unlike the majority group. This is because even though discovering τ objects from a group is enough for verifying that it is covered, one cannot *verify* a group is uncovered until there is a chance that the data set might still have enough objects from that group. Thus, assuming a non-zero probability for each unlabeled object to belong to each group, one might need to ask the crowd to label the entire data set before they can confirm that a specific group is uncovered.

Our idea for addressing this challenge is to design *a divide and conquer algorithm* that, instead of point queries, uses *set queries* to iteratively eliminate subsets of data that does not include any object from the given group. At a high level, our idea is to ask a set query from the crowd, inquiring whether the selected set contains at least one object from the given group $\mathbf{g}$. The user may provide two responses (yes/no). Interestingly, in either case, the user response provides valuable information that helps efficiently identify the coverage. If the answer is "No", the set does not include any object from the given group $\mathbf{g}$. As a result, the algorithm can safely prune the set, asking no further questions about it. In particular, for a group that is not covered, one can expect to see no answers on large set queries helping to prune a significant portion of the data set quickly. On the other hand, if the answer is "yes", the set contains at least one object from the group $\mathbf{g}$. As a result, the algorithm cannot prune the subset since it can have any number (larger than one) of the objects in $\mathbf{g}$. At first glance, the queries with yes answers do not provide helpful information as the algorithm cannot prune the subset (hence it needs to divide it into smaller subsets). However, a key observation is that the algorithm will only observe a limited number of yes answers before it stops. The reason is that the number of set queries with yes answers provides a lower-bound on the number of objects from $\mathbf{g}$ in the data set. As a result, the algorithm can stop as soon as the lower bound reaches τ, knowing that $\mathbf{g}$ is covered. The D&C approach verifies the data coverage for a given group, while our goal is to identify the uncovered regions for a given set of sensitive attributes. The next question is how to utilize this algorithm for efficient coverage identification on different scenarios of sensitive attributes, forming intersectional or non-intersectional groups. In particular, how can we find maximal uncovered patterns? Our idea is to apply sampling and aggregate estimation techniques to find the groups that even if merged are likely to still be uncovered. This will help reduce the coverage identification cost by running the D&C approach for the merged groups once.

data set	classifier	accuracy	precision on female
UTKFace: (`females`=200, `males`=2800)	DeepFace (opencv)	93.56	52.02
	DeepFace (retinaface)	94.16	56.15
	BaseCNN	97.6	74.8
UTKFace: (`females`=20, `males`=2980)	DeepFace (opencv)	96.53	**8.0**
	DeepFace (retinaface)	96.43	**10.09**
	BaseCNN	97.6	**21.59**

Figure 4: ML models' low performance for females in the presence of representation bias. [4]

3 Resolving Insufficient Representation

Data integration [5, 6] and data augmentation [7–10] are considered as the primary solutions for reducing data coverage issues in a data set. Data integration is promising when external sources of data are available. On the other hand, recent advancements in generative AI and foundation models have enabled efficient and effective augmentation of data sets with synthetic data. Therefore, in the following, we review two approaches, one from each category, in the context of lack of coverage resolution.

3.1 Data Integration

Data integration is to consolidate data from different sources into a single, unified view. Although it is an effective solution to acquire additional data from different distributions, there are sampling policy and cost-efficiency concerns that need to be examined. Therefore, *Data Distribution Tailoring (*DT*)* introduces data

integration techniques for resolving insufficient representation of subgroups in a data set in the most cost-effective manner [5]. A query to DT consists of a target schema, and a set of group distribution requirements in the form of the minimum counts (e.g., "`1,000 breast cancer monitoring data in Chicago with at least 30% label=positive, and at least 20% black patients`"). Collecting a fresh sample from a data view is costly (monetary, human resources, and/or computation cost) [11]. Therefore, DT focuses on satisfying the count requirements with minimum cost. Given an input query and a lake of available data sources, the first step is to discover a collection of candidate data views that satisfy the target schema. Each data view v_i is a projection-join $v_i = \Pi\big(D_{i1} \bowtie \cdots \bowtie D_{ik_i}\big)$, where D_{ij} is a data set in a given data lake. Let us suppose the data views are already discovered. At a high level, DT follows an iterative approach that at each iteration a data view is selected to be queried. Each query to a data view has a fixed cost and returns a sample that may or may not satisfy the query constraints. The samples that are either not fresh, or do not satisfy the query are discarded. Hence, the essential question towards a cost-effective data integration is *what data view to query next*. Depending on the available information about the data sources, various techniques may be employed.

For the cases when the group distributions are known, the process of collecting the target data set is a sequence of iterative steps, where at every step, the algorithm chooses a data view, queries it, and if the obtained tuple contributes to one of the groups for which the count requirement is not yet fulfilled, it is kept, otherwise discarded. To do so, a Dynamic Programming (DP) algorithm is proposed. An optimal source at each iteration minimizes the sum of its sampling cost plus the expected cost of collecting the remaining required groups, based on its sampling outcome. The DP algorithm, however, has a pseudo-polynomial time complexity. Hence, it quickly becomes intractable for cases where the minimum count requirements for the groups are not small. For cases where the (sensitive) attribute of interest is binary, such as (biological) `sex={male, female}`, and the cost to query data is similar from all sources, it turns out that the optimal strategy is to query the data source with maximum probability of obtaining a sample from the minority group. Expanding the binary-attributes algorithm for non-binary cases, the problem can be modeled as an extension of the "*coupon collector's*" problem [12], where the goal is to collect m_i instances from each coupon (group) $\mathbf{g}_i$. At each iteration, the coupon collector's algorithm identifies a data view as most promising and queries it. In simple terms, a data view with a smaller query cost and a higher chance of obtaining minority groups is more promising.

For the cases where the group distributions are unknown, we model DT as a *multi-armed bandit* problem, where every data view is modeled as an arm. Every arm has an unknown distribution of different groups while pulling an arm (i.e., querying the corresponding data view) has a cost. During various iterations, the algorithms pull the arms in an order that its expected total *reward* is maximized. Arguing that the reward of obtaining a tuple from a group is proportional to how rare this group is across different data views, we design the reward function based on the expected cost one needs to pay in order to collect a tuple from a specific group. As the bandit strategy, we adopt *Upper Confidence Bound (UCB)* to balance exploration and exploitation. At every iteration, for every arm, UCB computes confidence intervals for the expected reward and selects the arm with the maximum upper bound of reward to be explored next.

3.2 Data Augmentation using Foundation Models

While data integration provides a promising approach for resolving coverage issues in a data set, its effectiveness is limited to the availability of external data sources that are rich enough to find sufficient fresh samples from minority groups. This, however, is not always possible, especially since the minority samples are rare and not easy to obtain. Fortunately, recent advancements in Generative AI and Foundation Models have enabled synthesizing samples that are otherwise challenging to obtain from the real world.

Therefore, as an alternative approach to data integration, we turn our attention to the Foundation Models and Generative AI for resolving the lack of coverage. Particularly, models such as DALL.E[1] have emerged as

[1] `https://openai.com/dall-e-2`

powerful tools for generating multi-modal data such as image, audio, and video.

We formalize the foundation model $\mathcal{F}$ as a black-box function with the following inputs, that once queried synthesize an output tuple.

- **Prompt**: A natural language description providing instructions on the details of the tuple to be generated. For instance, a prompt for image generation might be "A realistic photo of a white cat running in a backyard."
- **Guide**: In cases where only a prompt is provided, the foundation model uses its imagination to generate the requested tuple. For the previous example, the prompt of a cat image, the breed, size, background, and other details are generated based on the model's imagination. Alternatively, a guide can be provided to influence the generation process. The guide is formalized as a pair (t, m) where t is a tuple and m is a mask specifying which parts of the guide tuple should be changed. Using the cat example, t can be a cat image and m can specify the foreground to be regenerated.

There are multiple challenges towards effective data set augmentations using foundation models. First, we have to determine the minimal set of synthetic tuples that once added to the original data set, under-representation issues are resolved. Second, the generated images should follow the underlying distribution represented in the input data set. Third, the generated tuples should have high quality and look realistic to a human evaluator. Last but not least, given the (often monetary) cost associated with the queries to the foundation model, we should ensure the cost-effectiveness of the data set repair process.

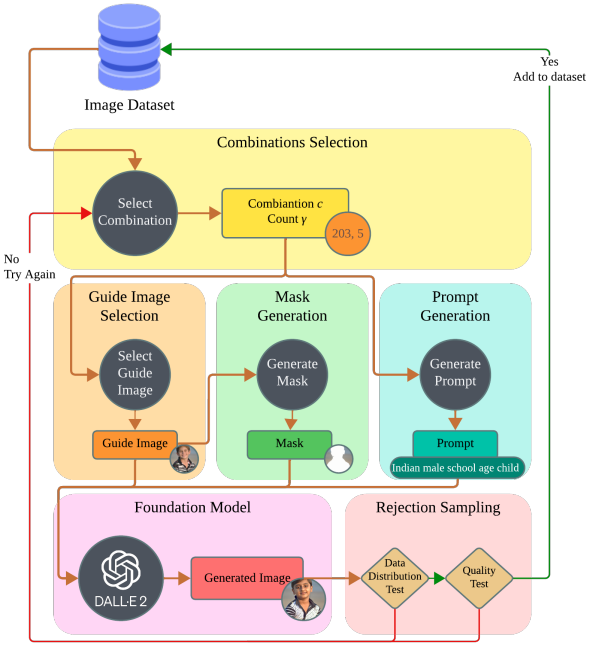


Figure 5: Architecture of CHAMELEON for image data augmentation for coverage enhancement.

Figure 5 shows the architecture of our system CHAMELEON [13] for coverage enhancement using DALL-E image generator. To address the first challenge, we define the combinations-selection problem, which minimizes the total number of synthetic tuples for resolving lack of coverage of minorities at the most general level. We show the problem is NP-hard, and propose a greedy approximation algorithm for it. To address the second and third challenges, CHAMELEON follows a *rejection sampling* strategy. It views each tuple in the data set $\mathcal{D}$ as an iid sample from the underlying distribution ξ it represents. It uses the vector representations (embeddings) space to describe the distribution. Then, given a newly generated tuple, it employs the one-class support vector machine (OCSVM) approach proposed by Scholkopf et al. [14] to reject the tuple if it does not follow ξ. Moreover, it models the quality evaluation as hypothesis testing and rejects the samples that have a higher chance of being labeled as "unrealistic" by a random human evaluator. Finally, to minimize the number of queries to the foundation model, we provide a guide tuple (and a mask), in addition to the prompt, to the foundation model. We model the guide-selection problem as *contextual multi-armed bandit* and propose a solution based on the contextual UCB for it.

Before concluding this section, let us provide some experiment results to demonstrate the effectiveness of data augmentation with CHAMELEON. We use FERET DB [15] for this experiment, which comprises 1199 individual images and serves as a standardized facial image database for researchers to develop algorithms and report results. All images in FERET DB share the same dimensions, pose, and facial expression. First, we identified the (level-1) uncovered ethnicity groups, using the threshold 80. We then used CHAMELEON and resolved the lack of coverage

Table 1: Illustrating the effect of lack of coverage repair using CHAMELEON on FERTDB

	Classifier Performance on FERTDB				Classifier Performance on Repaired			
Ethnicity Groups	#Images	Precision	Recall	F1-Score	#Images	Precision	Recall	F1-Score
Overall	756	0.81	0.75	0.78	987	0.70	0.75	0.72
Black	40	0.19	0.22	0.16	100	0.48	0.56	0.52
Hispanic	19	0.50	0.17	0.25	100	0.62	0.36	0.45
Middle Eastern	10	0.00	0.00	0.00	100	0.20	0.41	0.27

issues. To evaluate the effectiveness of the system, we trained a CNN model to predict the race of each image within this dataset. We then retrained the identical CNN on the repaired training data. Importantly, our test dataset for both experiments remains consistent and is derived from real images. Table 1 presents the improvements in precision, recall, and F1 score metrics for under-represented groups after repairing the dataset. The results indicate an enhancement in performance metrics for all under-represented groups following the repair process.

4 Generating Reliability Warnings

Interpretability is a necessity for data scientists who develop predictive models for critical decision-making. In such settings, it is important to provide additional means to support the following question: *is an individual prediction of the model reliable for decision-making?* Our goal is to use the lack of representation to help decision-makers find insights about this critical question. To further motivate this, let us use the following example:

Example 1: (Part1): Consider a judge who needs to decide whether to accept or deny a bail request. Using data-driven predictive models is prevalent in such cases for predicting recidivism [16]. Indeed, such models can be beneficial to help the judge make wise decisions. Suppose the model predicts the queried individual as high risk (or low risk). The judge is aware and concerned about the critics surrounding such models. A major question the judge faces is whether or not they should rely on the prediction outcome to take action for this case. Furthermore, if, for instance, they decide to ignore the outcome and hence they need to provide a statement supporting their action, what evidence can they provide?

In line with the recent trend on data-centric AI [17], we design novel approaches, complimentary to the existing work on trustworthy AI [18–21], to address the aforementioned trust question through the lens of *data*. In particular, unlike existing works that generate trust information from a *given model*, we associate *data sets with proper measurements* that specify their *the scope of use for predicting future cases*. We note that a predictive model provides only probabilistic guarantees on the average loss over the distribution represented by the data set used for training it. As a result, these predictions may not be distribution generalizable [22]. Consequently, if the query point is *not represented* by the data, the guarantees may not hold, hence one cannot rely on the prediction outcome. Besides, an essential requirement for a learning algorithm is that its training data $\mathcal{D}$ should represent the underlying distribution ξ. Even if so, the trained model h only provides a probabilistic guarantee on the expected loss on random samples from ξ. A model that performs well on *majority* of samples drawn from ξ will have a high performance on average. Still, as we observed in Figure 4, its performance for *minorities* and points that are not represented is questionable. Let us consider the following toy example:

Example 2: Consider a binary classification task where the input space is $\mathbf{x} = \langle x_1, x_2 \rangle$ and the output space is the binary label y with values $\{-1 \text{ (red)}, +1 \text{ (blue)}\}$. Suppose the underlying data distribution ξ follows a 2D Gaussian, where x_1 and x_2 are positively correlated as shown in Figure 6. The figure shows the data set $\mathcal{D}$ drawn

independently from the distribution ξ, along with their labels as their colors. Using $\mathcal{D}$, the prediction model h is constructed as shown in Figure 7. The decision boundary is specified in the picture; while any point above the line is predicted as +1, a query point below it is labeled as -1. The classifier has been evaluated using a test set that is an iid sample set drawn from the underlying data set ξ. The accuracy on the test set is high (above 90%), and hence, the model gets deployed. We cherry-picked four query points, $\mathbf{q}^1$ to $\mathbf{q}^4$, that are also included in Figure 7. Using h for prediction, $h(\mathbf{q}^1) = -1$, $h(\mathbf{q}^2) = +1$, $h(\mathbf{q}^3) = +1$, and $h(\mathbf{q}^4) = -1$. Figure 8 adds the ground-truth boundary to the search space, revealing the true label of the query points: every point inside the red circle has the true label -1 while any point outside of it is $+1$. Looking at the figure, $y^1 = +1$ while the model predicted it as $h(\mathbf{q}^1) = -1$. □

Let us take a closer look at the four query points in this example and their placement with regard to the tuples in $\mathcal{D}$ used for training h. $\mathbf{q}^2$ belongs to a *dense region* with many training tuples in $\mathcal{D}$ surrounding it. Besides, all of the tuples in its vicinity have the same label $y = +1$. As a result, one can expect that the model's outcome $h(\mathbf{q}^2) = +1$ should be a reliable prediction. Similar to $\mathbf{q}^2$, $\mathbf{q}^4$ also belongs to a dense region in $\mathcal{D}$; however, $\mathbf{q}^4$ belongs to an *uncertain region*, where some of the tuples in its vicinity have a label $y = +1$, and some others have the label $y = -1$. Considering the uncertainty in the vicinity of $\mathbf{q}^4$, one cannot confidently rely on the outcome of the model h. On the other hand, the neighbors of $\mathbf{q}^1$ (resp. $\mathbf{q}^3$) are not uncertain, all having the label $y = -1$ (resp. $y = +1$). However, the query points $\mathbf{q}^1$ and $\mathbf{q}^3$ are not well represented by $\mathcal{D}$. In other words, $\mathbf{q}^1$ and $\mathbf{q}^3$ are unlikely to be generated according to the underlying distribution ξ, represented by $\mathcal{D}$. As a result, following the no-free-lunch theorem [23], one cannot expect the outcome of model h to be reliable for these points. Looking at the ground-truth boundary in Figure 8, h luckily predicted the outcome for $\mathbf{q}^3$ correctly, but it was not fortunate to predict the y^1 correctly. Nevertheless, since the model is not reliably trained for these points, its outcome for these query points is not trustworthy.

From Example 2, we observe that the outcome of a model h, trained using a data set $\mathcal{D}$ is not reliable for a query point $\mathbf{q}$, if:

- **Lack of representation:** $\mathbf{q}$ is not well-represented by $\mathcal{D}$. In such cases, the model has not seen "enough" samples similar to $\mathbf{q}$ to reliably learn and predict the outcome of $\mathbf{q}$.
- **Lack of certainty:** $\mathbf{q}$ belongs to an uncertain region, where different tuples of $\mathcal{D}$ in the vicinity of $\mathbf{q}$ have different target values. $\mathbf{q}$ belongs to a high-fluctuating area, where tuples in the vicinity of $\mathbf{q}$ have a wide range of values.

Based on these two observations, we propose Representation-and-Uncertainty (**RU**) measures. To identify if a

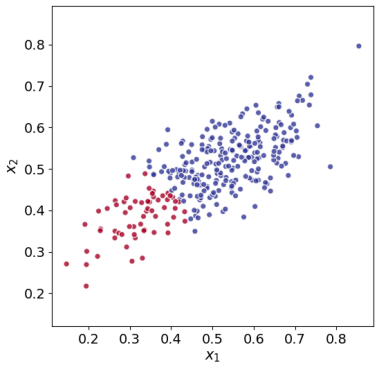


Figure 6: Data set $\mathcal{D}$ generated using a Gaussian distribution; x_1 and x_2 are positively correlated

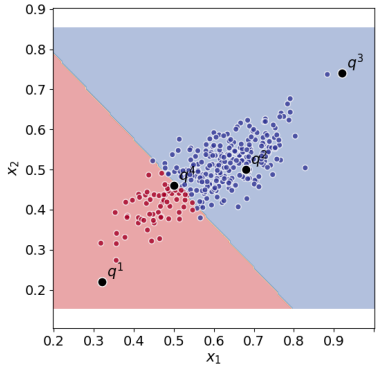


Figure 7: The decision boundary of learned model h and query points $\mathbf{q}^1$ to $\mathbf{q}^4$

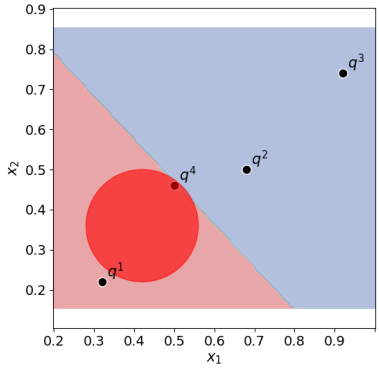


Figure 8: Ground-truth boundary, overlaid on the model decision boundary and query points

query suffers from uncertainty or lack of representation, one could use a deterministic approach using a fixed threshold. Then if the number of similar samples to (resp. label fluctuation in vicinity of) $\mathbf{q}$ is larger than the threshold it is considered as unrepresented (resp. uncertain). This approach, however, would be misleading since two numbers close to the threshold could be treated very differently. Also, all points on each side of the threshold would be considered equally represented (resp., certain). Instead, we consider *a randomized approach*, widely popular in the literature, including [24]. That is, instead of using fixed thresholds, a Bernoulli variable (a biased coin) is used that assigns $\mathbf{q}$ as unrepresented (resp., uncertain) based on the number of samples similar to it (resp., its neighborhood uncertainty). Given a query point $\mathbf{q}$, let $\mathbb{P}_o$ be the probability indicating if $\mathbf{q}$ is not represented and let $\mathbb{P}_u$ be the probability indicating if $\mathbf{q}$ belongs to an uncertain region. We represent the probability of the Bernoulli variables for lack of representation or uncertainty components as $\mathbb{P}_o$ and $\mathbb{P}_u$, respectively. Note that the two Bernoulli variables $\mathbb{P}_o$ and $\mathbb{P}_u$ are independent from each other. That simply follows the argument that after specifying the number of similar samples to $\mathbf{q}$ whether or not it should be considered as unrepresented does not depend on the uncertainty in the neighborhood of $\mathbf{q}$.

Definition 4.1 (STRONGRU) *The* STRONGRU *is a probabilistic measure that considers the outcome of a model for a query point* $\mathbf{q}$ *untrustworthy if* $\mathbf{q}$ *is not represented by* $\mathcal{D}$ *and it belongs to an uncertain region. Formally, the* STRONGRU *measure is:*

$$SRU(\mathbf{q}) = \mathbb{P}\big((\mathbf{q} \textit{ is outlier}) \wedge (\mathbf{q} \textit{ belongs to uncertain region})\big)$$

Since $\mathbb{P}_o$ *and* $\mathbb{P}_u$ *are independent:*
$$SRU(\mathbf{q}) = \mathbb{P}_o(\mathbf{q}) \times \mathbb{P}_u(\mathbf{q}) \tag{1}$$

STRONGRU raises the warning signal only when the query point fails on *both* conditions of being represented by $\mathcal{D}$ and not belonging to an uncertain region. For instance, in Example 2 none of the query points fail both on representation and on uncertainty; hence neither has a high STRONGRU score. On the other hand, a high STRONGRU score for a query point $\mathbf{q}$ *provides a strong warning signal* that one should perhaps reject the model outcome and not consider it for decision-making.

STRONGRU is a strong signal that raises warnings only for the fearfully concerning cases that fail both on representation and uncertainty. However, as observed in Example 2 a query points failing *at least* one of these conditions may also not be reliable, at least for critical decision making. We define the WEAKRU measure to raise a warning for such cases.

Definition 4.2 (WEAKRU) *The* WEAKRU *measure is a probabilistic measure that considers the outcome of a model for a query point* $\mathbf{q}$ *untrustworthy if* $\mathbf{q}$ *is not represented by* $\mathcal{D}$ **or** *it belongs to an uncertain region. Formally, the* WEAKRU *is computed as:*

$$WRU(\mathbf{q}) = \mathbb{P}\big((\mathbf{q} \textit{ is outlier}) \vee (\mathbf{q} \textit{ belongs to uncertain region})\big) = \mathbb{P}_o(\mathbf{q}) + \mathbb{P}_u(\mathbf{q}) - \mathbb{P}_o(\mathbf{q}) \times \mathbb{P}_u(\mathbf{q}) \tag{2}$$

Proposing quantitative probabilistic outcomes, RU measures are interpretable for the users, since beyond the scores, the uncertainty and lack of representation components provide an explanation to justify them. Please refer to [25] for more details on how to efficiently and effectively compute the representation ($\mathbb{P}_o$) and uncertainty ($\mathbb{P}_u$) probabilities, using only $\mathcal{D}$. In Example 1, let us see how the RU measures can be helpful.

Example 1. (part 2): *RU measures raise warning when the fitness of the data set used for drawing a prediction is questionable, helping the judge to be cautious when taking action. Besides, these measures provide quantitative evidence to support the judge's action when they decide to ignore a prediction outcome that is not trustworthy. The judge, for example, can argue to ignore a model outcome for a specific case, based on the insight that the model has been built using a data set that fails to represent the given case.* □

Finally, let us demonstrate the efficacy of RU measures through a series of experiments. Since the RU measures are *data-centric*, those are applicable for both classification and regression tasks, irrespective of the

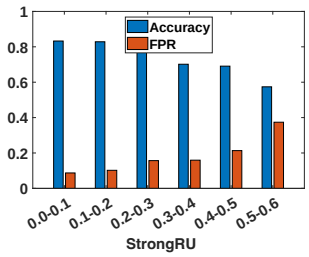


Figure 9: *Adult*, efficacy of STRONGRU on classification

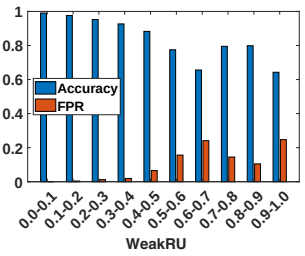


Figure 10: *Adult*, efficacy of WEAKRU on classification

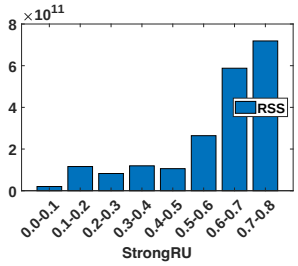


Figure 11: *House Sales in King County*, efficacy of STRONGRU on regression

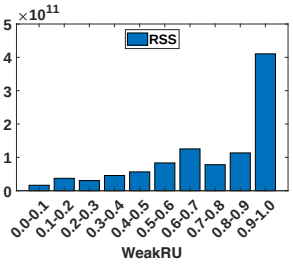


Figure 12: *House Sales in King County*, efficacy WEAKRU on regression

model used. We use *Adult* dataset [26] for classification and *House Sales in King County* dataset for the validation of regression tasks. From each dataset, we uniformly sample two sets from the underlying distribution. The first set serves as the training set to compute the RU values, and the second one is used as the test set from which the queries are drawn. We validate our proposal by providing the correlation between the RU values and the performance of an ML model's prediction on the same data.

We start by computing the RU values for all the query points in the test set. Next, we bucketize the query points based on their RU values in equi-width buckets of width 0.1. We repeat this for both STRONGRU and WEAKRU measures. Next, we train a model on the training data set and predict the target variable for the points in each range of RU measure. The validation results for the classification task on the *Adult* dataset are presented in Figures 9 and 10. Each figure corresponds to the accuracy/error measures of the classifier over each bucket of RU values for STRONGRU and WEAKRU. As the RU values increase, the accuracy of the model drops while the FPR rises, and therefore, the model fails to capture the ground truth for the points that fall into untrustworthy regions in the data set. By repeating the aforementioned steps for the regression task on the *House Sales in King County* dataset, we observe similar results presented in Figures 11 and 12. As the RU value increases, the RSS of the regression model follows the same trend denoting that the model fails to perform for tuples with a high RU value.

5 Related Work

Bias in data has been looked at for a long time in statistical community [27] but social data presents different challenges [28–32]. The diversity and representativeness of data have been widely studied [32], in fields such as social science [33–35], political science [36], and information retrieval [37]. Tracing back machine bias to its source, there have been major efforts to identify different types [28, 38, 39] and sources [40–42] of biases in data. Efforts to satisfy *responsible data* requirements [6] extend to various stages of the data analysis pipeline, including data annotation [43, 44], data cleaning and repair [45–47], data imputation [48], entity resolution [49, 50], data integration [5, 6], etc.

Data Coverage: The notion of data coverage has received extensive attention from different angles. Detecting lack of coverage has been studied for datasets with discrete [1] and continuous [3] attributes populated in single or multiple [51] relations. To resolve insufficient coverage, [52–54] consider resolving representation bias in preprocessing pipelines by rewriting queries into the closest operation so that certain subgroups are sufficiently represented in the downstream tasks. Alternatively, [1, 55] propose a data collection strategy to acquire as little additional data as possible (to minimize the associated costs) to meet the representation constraints. [7, 9, 10] opt for a data augmentation approach by adding partially altered duplicates of already existing tuples or generating new synthetic entries from existing data. Consequently, the new data set has an equal number of elements for different groups, resulting in potentially resolving the under-representation issues. Finally, [5] utilizes data

integration techniques to consolidate data from different sources into a single dataset to resolve representation bias. Related works also include [55–57] that seek to understand if the overall performance of the model fails to reflect and performs poorly on certain slices in the data. As alternative approaches to measure representation bias, the notion of representation rate [10] (a.k.a. equal base rate [58]) is introduced which compared with coverage, it is more restrictive as it requires almost equal ratios from different groups. Please refer to [2] for a comprehensive survey about representation bias in data.

ML Reliability: Model-centric works for uncertainty quantification such as probabilistic classifiers [59–62], prediction intervals (PIs) [63–65] and conformal predictions (CP) [66, 67] that are used for measuring prediction uncertainty, are built by maximizing the *expected performance* on *random* sample from the underlying distribution. As a result, while providing accurate estimations for the dense regions of data (e.g. majority groups), their estimation accuracy is questionable for the poorly represented regions. In particular, [66] recognizes the lack of guarantees in the performance of CP for such regions. Besides, the bulk of work on trustworthy AI provides information that *supports* the outcome of an ML model. For example, existing work on explainable AI, including [68–70], aims to find simple explanations and rules that justify the outcome of a model. Conversely, we aim to *raise warning signals* when the outcome of a model is *not* trustworthy. That is, to provide reasons that *cast doubt* on the reliability of the model outcome for a given query point.

6 Final Remarks

As Data-centric AI and Responsible AI emerge as focal points in data science research, the development of Data-centric methodologies for ensuring Responsible and Trustworthy AI attracts increasing attention. While there is some excellent work on responsible data management to achieve this goal, there remain many challenges yet to be addressed.

In this paper, we focused on a crucial aspect of responsible data – detecting and addressing the under-representation of minorities within a data set. We formally defined the notion of data coverage and discussed various techniques for (a) identifying lack of representation issues across different data modalities, (b) ensuring proper representation of minorities in data, and (c) limiting the scope-of-use of data sets based on their representation issues by generating proper (RU) warning signals. Even though the research on detecting lack of coverage issues is relatively mature, resolution techniques are still understudied. Considering the recent advancements in Generative AI, utilizing Foundation Models and Large Language Models, and studying their limitations, for data augmentation to improve the representation of minorities at the data level seems interesting to further explore.

References

[1] A. Asudeh, Z. Jin, and H. Jagadish. Assessing and remedying coverage for a given dataset. In ICDE, pages 554–565. IEEE, 2019.

[2] N. Shahbazi, Y. Lin, A. Asudeh, and H. Jagadish. Representation bias in data: A survey on identification and resolution techniques. ACM Computing Surveys, 2023.

[3] A. Asudeh, N. Shahbazi, Z. Jin, and H. V. Jagadish. Identifying insufficient data coverage for ordinal continuous-valued attributes. In SIGMOD. ACM, 2021.

[4] M. Mousavi, N. Shahbazi, and A. Asudeh. Data coverage for detecting representation bias in image datasets: A crowdsourcing approach. In EDBT, pages 47–60, 2024.

[5] F. Nargesian, A. Asudeh, and H. Jagadish. Tailoring data source distributions for fairness-aware data integration. Proceedings of the VLDB Endowment, 14(11):2519–2532, 2021.

[6] F. Nargesian, A. Asudeh, and H. V. Jagadish. Responsible data integration: Next-generation challenges. SIGMOD, 2022.

[7] S. Sharma, Y. Zhang, J. M. Ríos Aliaga, D. Bouneffouf, V. Muthusamy, and K. R. Varshney. Data augmentation for discrimination prevention and bias disambiguation. In AIES, pages 358–364, 2020.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res., 16:321–357, 2002.

[9] V. Iosifidis and E. Ntoutsi. Dealing with bias via data augmentation in supervised learning scenarios. Jo Bates Paul D. Clough Robert Jäschke, 24, 2018.

[10] L. E. Celis, V. Keswani, and N. Vishnoi. Data preprocessing to mitigate bias: A maximum entropy based approach. In ICML, pages 1349–1359. PMLR, 2020.

[11] A. Asudeh and F. Nargesian. Towards distribution-aware query answering in data markets. Proceedings of the VLDB Endowment, 15(11):3137–3144, 2022.

[12] R. Motwani and P. Raghavan. Randomized algorithms. Cambridge university press, 1995.

[13] M. Erfanian, H. V. Jagadish, and A. Asudeh. Chameleon: Foundation models for fairness-aware multi-modal data augmentation to enhance coverage of minorities. arXiv preprint arXiv:2402.01071, 2024.

[14] B. Schölkopf, R. C. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt. Support vector method for novelty detection. NeurIPS, 12, 1999.

[15] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss. The feret database and evaluation procedure for face-recognition algorithms. Image and vision computing, 16(5):295–306, 1998.

[16] J. Dressel and H. Farid. The accuracy, fairness, and limits of predicting recidivism. Science advances, 4(1):eaao5580, 2018.

[17] A. Ng. Mlops: From model-centric to data-centric AI. 2021.

[18] J. M. Wing. Trustworthy AI. CACM, 64(10):64–71, 2021.

[19] M. Kentour and J. Lu. Analysis of trustworthiness in machine learning and deep learning. InfoComp, 2021.

[20] H. Liu, Y. Wang, W. Fan, X. Liu, Y. Li, S. Jain, A. K. Jain, and J. Tang. Trustworthy AI: A computational perspective. arXiv preprint arXiv:2107.06641, 2021.

[21] R. Singh, M. Vatsa, and N. Ratha. Trustworthy AI. In 8th ACM IKDD CODS and 26th COMAD, pages 449–453. 2021.

[22] B. Kulynych, Y.-Y. Yang, Y. Yu, J. Błasiok, and P. Nakkiran. What you see is what you get: Distributional generalization for algorithm design in deep learning. arXiv preprint arXiv:2204.03230, 2022.

[23] S. M. Kakade. On the sample complexity of reinforcement learning. University of London, University College London (United Kingdom), 2003.

[24] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In ITCS, pages 214–226, 2012.

[25] N. Shahbazi and A. Asudeh. Data-centric reliability evaluation of individual predictions. CoRR, abs/2204.07682, 2022.

[26] M. Lichman. Adult income dataset, UCI machine learning repository. `https://archive.ics.uci.edu/ml/datasets/adult`, 2013.

[27] J. Neyman and E. S. Pearson. Contributions to the theory of testing statistical hypotheses. Statistical Research Memoirs, 1936.

[28] A. Olteanu, C. Castillo, F. Diaz, and E. Kiciman. Social data: Biases, methodological pitfalls, and ethical boundaries. Frontiers in Big Data, 2:13, 2019.

[29] S. Barocas, M. Hardt, and A. Narayanan. Fairness and machine learning: Limitations and opportunities. `fairmlbook.org`, 2019.

[30] S. Barocas and A. D. Selbst. Big data's disparate impact. Calif. L. Rev., 104:671, 2016.

[31] J. Kleinberg. Fairness, rankings, and behavioral biases. FAT*, 2019.

[32] M. Drosou, H. Jagadish, E. Pitoura, and J. Stoyanovich. Diversity in big data: A review. Big data, 5(2):73–84, 2017.

[33] E. Berrey. The enigma of diversity: The language of race and the limits of racial justice. University of Chicago Press, 2015.

[34] F. Dobbin and A. Kalev. Why diversity programs fail and what works better. Harvard Business Review, 94(7-8):52–60, 2016.

[35] E. H. Simpson. Measurement of diversity. Nature, 163(4148), 1949.

[36] J. Surowiecki. The wisdom of crowds. Anchor, 2005.

[37] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In WSDM, pages 5–14. ACM, 2009.

[38] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. ACM Computing Surveys (CSUR), 54(6):1–35, 2021.

[39] B. Friedman and H. Nissenbaum. Bias in computer systems. TOIS, 14(3):330–347, 1996.

[40] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In CVPR 2011, pages 1521–1528. IEEE, 2011.

[41] K. Crawford. The hidden biases in big data. Harvard business review, 1(4), 2013.

[42] N. Diakopoulos. Algorithmic accountability: Journalistic investigation of computational power structures. Digital journalism, 3(3):398–415, 2015.

[43] Y. Li, H. Sun, and W. H. Wang. Towards fair truth discovery from biased crowdsourced answers. In SIGKDD, pages 599–607, 2020.

[44] S. Lazier, S. Thirumuruganathan, and H. Anahideh. Fairness and bias in truth discovery algorithms: An experimental analysis. arXiv preprint arXiv:2304.12573, 2023.

[45] B. Salimi, L. Rodriguez, B. Howe, and D. Suciu. Interventional fairness: Causal database repair for algorithmic fairness. In SIGMOD, pages 793–810. ACM, 2019.

[46] K. H. Tae, Y. Roh, Y. H. Oh, H. Kim, and S. E. Whang. Data cleaning for accurate, fair, and robust models: A big data-AI integration approach. In DEEM workshop, pages 1–4, 2019.

[47] B. Salimi, B. Howe, and D. Suciu. Database repair meets algorithmic fairness. ACM SIGMOD Record, 49(1):34–41, 2020.

[48] F. Martínez-Plumed, C. Ferri, D. Nieves, and J. Hernández-Orallo. Fairness and missing values. arXiv preprint arXiv:1905.12728, 2019.

[49] N. Shahbazi, N. Danevski, F. Nargesian, A. Asudeh, and D. Srivastava. Through the fairness lens: Experimental analysis and evaluation of entity matching. Proceedings of the VLDB Endowment, 16(11):3279–3292, 2023.

[50] N. Fanourakis, C. Kontousias, V. Efthymiou, V. Christophides, and D. Plexousakis. Fairer demo: Fairness-aware and explainable entity resolution. 2023.

[51] Y. Lin, Y. Guan, A. Asudeh, and H. Jagadish. Identifying insufficient data coverage in databases with multiple relations. Proceedings of the VLDB Endowment, 13(12):2229–2242, 2020.

[52] C. Accinelli, S. Minisi, and B. Catania. Coverage-based rewriting for data preparation. In EDBT Workshops, 2020.

[53] C. Accinelli, B. Catania, G. Guerrini, and S. Minisi. The impact of rewriting on coverage constraint satisfaction. In EDBT Workshops, 2021.

[54] S. Shetiya, I. P. Swift, A. Asudeh, and G. Das. Fairness-aware range queries for selecting unbiased data. In ICDE. IEEE, 2022.

[55] K. H. Tae and S. E. Whang. Slice tuner: A selective data acquisition framework for accurate and fair machine learning models. In SIGMOD, pages 1771–1783, 2021.

[56] Y. Chung, T. Kraska, N. Polyzotis, K. H. Tae, and S. E. Whang. Slice finder: Automated data slicing for model validation. In ICDE, pages 1550–1553. IEEE, 2019.

[57] S. Sagadeeva and M. Boehm. Sliceline: Fast, linear-algebra-based slice finding for ml model debugging. In SIGMOD, pages 2290–2299, 2021.

[58] J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores. arXiv preprint arXiv:1609.05807, 2016.

[59] B. Zadrozny and C. Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In ICML, volume 1, pages 609–616. Citeseer, 2001.

[60] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In SIGKDD, pages 694–699, 2002.

[61] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in large margin classifiers, 10(3):61–74, 1999.

[62] A. Niculescu-Mizil and R. Caruana. Predicting good probabilities with supervised learning. In Proceedings of the 22nd international conference on Machine learning, pages 625–632, 2005.

[63] C. Chatfield. Prediction intervals. Journal of Business and Economic Statistics, 11:121–135, 1993.

[64] T. Pearce, A. Brintrup, M. Zaki, and A. Neely. High-quality prediction intervals for deep learning: A distribution-free, ensembled approach. In International conference on machine learning, pages 4075–4084. PMLR, 2018.

[65] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya. Lower upper bound estimation method for construction of neural network-based prediction intervals. IEEE transactions on neural networks, 22(3):337–346, 2010.

[66] A. N. Angelopoulos and S. Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. arXiv preprint arXiv:2107.07511, 2021.

[67] G. Shafer and V. Vovk. A tutorial on conformal prediction. Journal of Machine Learning Research, 9(3), 2008.

[68] M. Harradon, J. Druce, and B. Ruttenberg. Causal learning and explanation of deep neural networks via autoencoded activations. arXiv preprint arXiv:1802.00541, 2018.

[69] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In SIGKDD, pages 1135–1144, 2016.

[70] D. Gunning and D. Aha. Darpa's explainable artificial intelligence (XAI) program. AI Magazine, 40(2):44–58, 2019.

Overcoming Data Biases: Towards Enhanced Accuracy and Reliability in Machine Learning

Jiongli Zhu, Babak Salimi
University of California San Diego
{jiz143, bsalimi}@ucsd.edu

Abstract

The pervasive integration of machine learning (ML) across various sectors has underscored the critical challenge of addressing inherent biases in ML models. These biases not only undermine the models' fairness and accuracy but also have significant real-world consequences. Traditional approaches to mitigating these biases often fail to address their root causes, leading to solutions that may superficially seem fair but do not tackle the underlying problems. This review paper explores the role of causal modeling in enhancing data cleaning, preparation, and quality management for ML. By analyzing existing research, we demonstrate how causal reasoning can effectively identify and rectify data biases, thus improving the fairness and accuracy of ML models. We advocate for the increased adoption of causal approaches in these processes, emphasizing their potential to significantly enhance the integrity and reliability of data-driven technologies.

1 Introduction

Machine Learning has become integral to sectors such as healthcare, finance, and law enforcement, spotlighting the importance of addressing biases and inaccuracies in ML models. These critical issues necessitate the development of ML models that are reliable, accurate, and fair, given their significant impact on individuals and communities. Consequently, substantial research efforts have been dedicated to mitigating algorithmic bias, aiming to enhance the robustness, reliability, accuracy, and fairness of ML models [3, 57].

Despite numerous efforts to address data biases in ML, current strategies often focus on alleviating the symptoms rather than confronting the underlying causes of these biases. This approach may inadvertently lead to "fair-washing," where superficial measures worsen the problems they intend to solve [96]. In the realm of developing fair ML models, prevalent methods include: (1) integrating fairness metrics into the optimization process during training, known as in-processing [10, 12, 42, 44, 88, 89], and adjusting the model's output post-training, referred to as post-processing [34, 41, 67, 82]; and (2) modifying the data before training, or pre-processing, to achieve a more balanced distribution [11, 26, 40, 74, 87]. However, these approaches often operate under the assumption that the training data is representative of the actual distribution [37], a premise that is frequently flawed. Data biases, such as confounding, measurement, and selection biases, along with other data quality issues, distort the data distribution [13, 27, 57, 61, 62, 96], often leading to training datasets that do not

Copyright 2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

accurately represent the target population. This mismatch poses challenges in preprocessing the data to obtain a representative sample. Consequently, ML models trained with such biased data are likely to underperform, being unfair and inaccurate when applied to the target population and during the inference process.

Considerable efforts have been directed toward mitigating data biases, including selection bias [17, 20, 36, 52, 70] and labeling errors [39, 90, 93]. Yet, these initiatives often hinge on significant assumptions—like the presence of an unbiased sample or specific presumptions regarding data quality issues—that are challenging to verify in real-world settings. Such reliance introduces complications, rendering these strategies less effective for practical applications. Furthermore, traditional data cleaning techniques [43, 48, 56, 69] may falter in restoring the ground truth or in ensuring datasets accurately reflect their target domains. Occasionally, it may be inherently unfeasible to obtain a representative sample to efficiently counter data biases. Overlooking these pivotal concerns may inadvertently perpetuate existing biases within the data [31, 61, 75].

In this paper, we explore data biases through a causal lens, integrating concepts from ML , causal inference, and data management. Our primary objective is to highlight the significant potential of causal reasoning in enhancing data cleaning techniques, with a particular focus on data quality management research. Causal reasoning facilitates a more thorough examination and validation of the assumptions underlying data collection processes and data provenance, thereby increasing transparency. By reviewing recent studies that employ causal inference for debiasing data, we aim to showcase the considerable impact of this methodology. Our analysis focuses on incorporating causal methods into existing frameworks for data quality management and cleaning, with the goal of reducing biases and improving both the fairness and accuracy of ML models. This specific investigation contributes to the expanding field of data quality management research, an essential component of data management. We advocate for ongoing research and development aimed at forging more robust, unbiased, and effective data-driven technologies, achieved through the refinement of data management practices.

2 Data Biases

Data bias refers to systematic errors within datasets that lead to outcomes that are either inaccurate, unfair, or unreliable. These biases often manifest as uncertainties and incompleteness in data and systematic deviations in the data distribution, compromising its representation of the actual phenomena under study. In the context of knowledge extraction, these biases can lead to analyses that yield incorrect conclusions and false discoveries. In the context of ML, if these biases are not adequately addressed, they can be learned and perpetuated by downstream models, impacting their accuracy and fairness. In this section, we explore the most common sources of data bias in real-world applications, with a specific focus on challenges such as bias due to missing data, confounding variables, and erroneous measurements [5, 28, 57, 65, 95]. Understanding and addressing these factors is vital for assessing the quality and reliability of data used in ML.

Bias due to Missing Data: During data collection, certain portions of data may be missing for various reasons, such as high data collection costs for specific sub-populations or historical discrimination [1]. This missing data can manifest as either *missing values within tuples* or *entirely missing tuples*. It is particularly challenging when entire tuples are missing non-randomly, leading to selection bias [15]. This bias occurs when the data collection process or the selection of training data is influenced by specific attributes, resulting in a subset that does not accurately represent the entire population. Even in scenarios where recovery is theoretically possible [1], such as cases of data missing completely at random or when an unbiased sample is available, the existing approaches for dealing with missing data imputation or selection bias typically only provide asymptotic guarantees [45, 72, 76]. In practical applications with finite data, these methods might display unpredictable behavior and still lead to biased samples. The non-random nature of missing data thus presents significant challenges in obtaining unbiased

[1] Recoverability of a distribution from missing data, biases, or data quality issues, in principle, refers to the capacity to accurately and consistently (asymptotically) estimate the underlying probability distribution or statistical properties of a dataset, even in the presence of such data quality challenges.

data that accurately reflects the underlying distribution, highlighting a crucial concern in data management and ML. We illustrate with two examples:

Example 1 (Missing Attribute Values) *Missing data presents a formidable challenge in critical areas such as healthcare and finance, characterized by its non-random occurrence and complex mechanisms. In pediatric health studies, for example, in cancer research, parents' hesitancy to divulge sensitive prognosis details, like life expectancy, results in crucial information being omitted. Studies have shown that such omissions correlate with poorer survival outcomes in comprehensive cancer registries [68, 71]. Similarly, financial ML applications face missing data, particularly in loan application datasets, where information on repayment potential for rejected applicants or those with restricted financial access is often absent. This gap, largely due to historical and racial biases, distorts data distribution. If not addressed, this distortion leads to inaccurate estimations and perpetuates biases in these sectors [22, 54, 68].*

Example 2 (Selection Bias) *Selection bias is prevalent in many sensitive domains, such as health care, finance, and predictive policing. In predictive policing, selection bias may occur when historical crime data, which often reflects past law enforcement and societal biases, is used to train ML models. This can lead to a cycle where certain communities are over-policed based on biased data, further perpetuating the bias in future decision makings [9, 53]. In covid-19 studies, selection bias can arise when the data is collected from a population of individuals who are hospitalized or have tested positive, leading to a false association between the test positive rate and ethnic minorities due to barriers in healthcare access [30]. In finance, selection bias can manifest in credit scoring where historical lending data may disproportionately represent certain socio-economic groups, such as individuals from higher income areas. This can lead to unfair or inaccurate credit decisions when the model is applied to populations from diverse economic backgrounds, including underdeveloped regions [4, 80].*

Bias due to Latent Confounding: Confounding bias arises in ML when unobserved confounders affect both predictors and outcomes, leading to spurious correlations and misinterpreted causal relationships [64]. This bias can distort conclusions, making data associations that seem causal when they are not. In ML, models trained on such data may base predictions on these unreliable correlations, resulting in inaccuracies and poor generalization across real-world scenarios [2, 35, 84].

Example 3 (Confounding Bias) *Confounding bias significantly impacts ML applications in healthcare and social media analytics. In healthcare, for instance, ML models trained on skin cancer images may falsely associate surgical markings with disease severity, misguiding the diagnosis [23, 81]. Similarly, pneumonia detection models may inaccurately correlate device fingerprints with the disease by using data pooled from hospitals with varying pneumonia rates, leading to misidentifications based on hospital systems rather than the disease itself [86]. In social media analytics, complex relationships between various factors and self-harm tendencies create biased associations between social media use and self-harm, complicating the analysis [79].*

Bias due to Measurement Error: Measurement errors arise when there is a discrepancy between the true value of a variable and the value obtained through measurement or observation. When these errors are not random but systematically affect certain sub-populations, this results in skewed data distribution, a situation known as measurement bias [49, 58, 65]. A prevalent form of measurement bias is label bias. Label bias arises when irrelevant factors, such as sensitive demographic information, influence the assigned labels during the data collection process.

Example 4 (Measurement Bias) *In epidemiological studies estimating cardiovascular risk from dietary habits, reliance on self-reported dietary intake questionnaires can introduce measurement bias. Participants often*

misreport consumption—understating unhealthy and overstating healthy foods due to social desirability—skewing data away from true dietary patterns. This misalignment can lead models to underestimate the benefits of healthy diets on heart disease prevention [63]. Similarly, in computer vision or natural language processing, crowdsourced data labeling can embed label bias. For example, facial recognition models may perform poorly on certain ethnic groups if labels are influenced by unconscious stereotypes, undermining the model's accuracy and fairness in applications like surveillance [33].

3 Causal Modeling of Data Biases

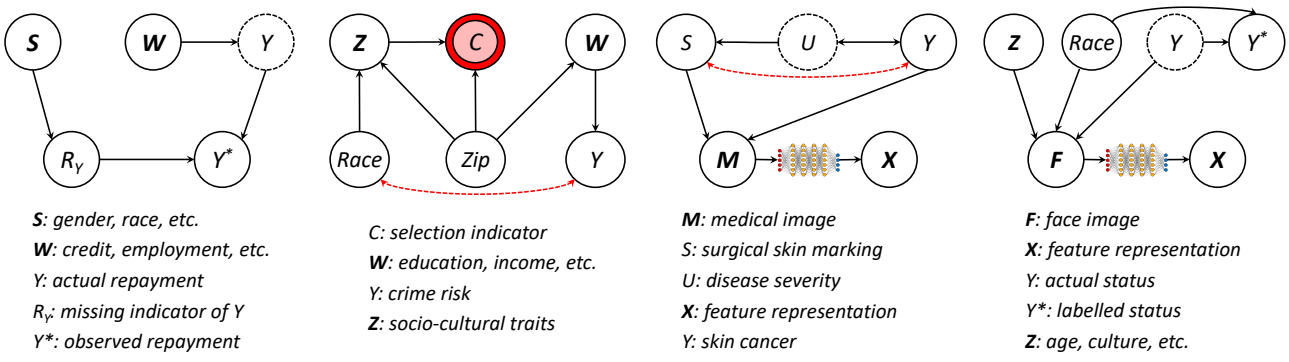
In this section, we demonstrate the essential role of causal modeling in addressing various data biases. Causal modeling provides a structured framework for understanding and capturing the provenance of data collection processes, along with their intricacies. This approach is crucial in identifying the sources of bias and plays a key role in informing the development and implementation of data debiasing and cleaning algorithms. By leveraging causal relationships, these algorithms are better equipped to tackle the root causes of bias, rather than merely addressing their symptoms. Such an approach leads to the creation of a more robust and reliable dataset, which is vital for building fair and accurate ML models.

Causal Diagrams: A causal diagram or causal graph is a directed graph that represents the causal relationships between a collection of observed or unobserved (latent) variables and models the underlying process that generates the observed data. Each node in a causal diagram corresponds to a variable, and an edge between two nodes indicates a potential causal relationship between the two variables. To illustrate, consider the causal diagram shown in Figure 1b. In this graph, the edge from the various factors such as education and income ($\boldsymbol{W}$) to the crime risk (Y) indicates that these factors of a person causally influence their risk of committing crimes.

d-separation and Conditional Independence: Causal diagrams encode a set of conditional independences that can be read off the graph using d-separation [64]. Two nodes are d-separated by a set of variables $\boldsymbol{V}_m$ in causal diagram G, denoted $(V_l \perp\!\!\!\perp V_r \mid_d \boldsymbol{V}_m)$ if for every path between them, one of the following conditions holds: (1) the path contains a chain ($V_l \to V \to V_r$) or a fork ($V_l \leftarrow V \to V_r$) such that $V \in \boldsymbol{V}_m$, and (2) the path contains a collider ($V_l \to V \leftarrow V_r$) such that $V \notin \boldsymbol{V}_m$, and no descendants of V are in $\boldsymbol{V}_m$. A distribution is said to be Markov compatible with a causal graph if d-separation within the graph implies conditional independence in the data distribution, i.e., $(V_l \perp\!\!\!\perp V_r \mid_d \boldsymbol{V}_m) \Rightarrow (V_l \perp\!\!\!\perp V_r \mid \boldsymbol{V}_m)$. Continuing with the causal diagram in Figure 1b, the graph encodes the d-separation statement $(Y \perp\!\!\!\perp \text{Zip} \mid_d \boldsymbol{W})$. For any distribution that is Markov compatible with this graph, this d-separation implies that crime risk (Y) and neighborhood (Zip) are independent, conditioned on education and income ($\boldsymbol{W}$). In this paper, assuming Markov compatibility, we consider d-separation to always imply conditional independence and use these terms interchangeably.

Next, we model each of the data biases using causal diagrams. Our discussion primarily centers on three specific types of biases: non-random missing values and selection bias as instances of bias due to missing data, confounding bias resulting from variable omission, and label bias as a manifestation of measurement errors. In addition, we explore existing research that addresses various forms of data biases in ML applications and discuss recent works that utilize the conditional independences encoded in causal diagrams for building fair ML models.

Algorithmic Fairness: Fairness in ML centers around a model h producing an output $h(\boldsymbol{x})$ and considering a protected attribute S, like gender or race. Many existing definitions of fairness require some form of statistical independence between the model's output and the protected attribute, which is sometimes conditioned on a third set of variables [57]. For instance, *statistical parity* ([21]) necessitates equal positive and negative prediction rates across different groups, formalized as $(S \perp\!\!\!\perp h(\boldsymbol{x}))$. *Equalized odds* ([34]) aims for parity in false positive and negative rates across groups, denoted as $(S \perp\!\!\!\perp h(\boldsymbol{x}) \mid Y)$. Meanwhile, *conditional statistical parity* seeks consistent positive classification probabilities across groups when accounting for certain permissible attributes $\boldsymbol{A}$, which are considered non-discriminatory factors in decision-making, expressed as $(S \perp\!\!\!\perp h(\boldsymbol{x}) \mid \boldsymbol{A})$. Notably,



(a) (b) (c) (d)

Figure 1: (a) A causal diagram for modeling missing values in pediatric health studies. (b) A causal diagram for modeling selection bias in predictive policing. (c) A causal diagram for modeling confounding bias in medical imaging. (d) A causal diagram for modeling label bias in facial image labeling. The elements of a causal diagram are: ○ representing observed attributes, ⋯ denoting unobserved attributes, → illustrating a causal dependency between two variables, ↔ indicating a correlation due to common parent variables not included in the diagram, and ⋯→ signifying a spurious correlation due to data biases.

conditional statistical parity is a more general fairness concept compared to the others. When the set of admissible features $\boldsymbol{A}$ is empty, it simplifies to statistical parity, and when $\boldsymbol{A}$ includes the outcome label Y, it becomes equivalent to equalized odds. Many other associational and causal fairness criteria can also be expressed as conditional independence constraints [73].

3.1 Modeling Bias due to Missing Data

3.1.1 Missing Values

Missing values within a variable U can be efficiently represented using a binary missing indicator variable R_U, which denotes the presence or absence of data in U. Specifically, $R_U = 1$ denotes a non-missing (present) value, whereas $R_U = 0$ indicates a missing value. Furthermore, let U^* denote the observed dataset from U, where missing entries are filled with a placeholder (e.g., null). We assume that only U is subject to missing values, with other variables' actual values being completely observed. The interaction between U, R_U, and U^* can be formally depicted as follows:

$$U^* = \begin{cases} U, & \text{if } R_U = 1, \\ \text{null}, & \text{if } R_U = 0. \end{cases}$$

The subset of data that contains no missing values can be considered a sample from the distribution $\Pr(\boldsymbol{V} \mid R_U = 1)$, where $\boldsymbol{V}$ denotes the set of all variables. This implies that the subset of data without missing values is representative of the underlying distribution only if $\Pr(\boldsymbol{V} \mid R_U = 1) = \Pr(\boldsymbol{V})$. The condition for this equality is that the occurrence of missing values is completely random, denoted as $(R_U \perp\!\!\!\perp \boldsymbol{V})$, which suggests that R_U is causally independent of all other variables. However, in cases where missingness is not random, R_U is causally influenced by other variables. Such influence can be depicted in a causal graph with edges from influencing variables to R_U, thus capturing the missingness pattern. Causal modeling, therefore, provides a comprehensive framework to explicitly identify the sources of non-random missing values and to understand their effects on the data distribution. It also aids in studying the sufficient and necessary conditions for the recoverability of missing data, thereby enhancing the robustness and applicability of data analysis in various contexts.

Next, we use a concrete example in credit risk assessment to show how non-random missing values can be modeled using causal diagrams and how this would affect the downstream ML model.

Example 5 (Credit Risk Assessment) *The causal diagram in Figure 1a depicts the scenario of missing values in loan application data, as discussed in Example 1. This graph indicates that the actual loan repayment label Y is independent of demographic factors $\boldsymbol{S}$, i.e., ($\boldsymbol{S} \perp\!\!\!\perp Y$), suggesting that demographic information ($\boldsymbol{S}$) does not correlate with loan repayment (Y) in the underlying distribution ($\Pr(Y \mid \boldsymbol{S}) = \Pr(Y)$). The observed version of Y, denoted as Y^*, exhibits missingness influenced by individual demographics, reflecting that records from certain demographic groups are more prone to incompleteness due to historical biases. This relationship is captured in the causal diagram by the missingness variable R_Y, which is dependent on the demographic information $\boldsymbol{S}$. Given the high correlation between the occurrence of missing values and demographics, any imputation method with errors could lead to a biased dataset. Consequently, the imputed labels Y^*_{imp} could become strongly correlated with demographic factors $\boldsymbol{S}$. This outcome demonstrates the challenges in handling missing data, particularly when such missingness is non-randomly linked with demographic attributes. Consequently, models trained on this observed data are likely to be unfair, perpetuating historical biases.*

Data imputation methods in practice often assume that missing data occurs either completely at random (MCAR) or at random (MAR), which suggests that the mechanism of missingness does not depend on the actual values of the variable that is missing [29]. However, these methods may introduce bias when the missingness mechanism is not at random (MNAR), meaning the missingness of a variable is influenced by its own actual values or other latent variables. Such conditions render traditional imputation strategies prone to producing biased data as the original, true values of the data are typically not recoverable [29, 32, 50, 66]. Consequently, ML models trained on this biased, imputed data inherit and perpetuate the bias, leading to unfair and unreliable outcomes. To mitigate these challenges, causal modeling has been instrumental in identifying both the necessary and sufficient conditions for effectively recovering from data missingness. Additionally, it aids in pinpointing which statistics or parts of the distribution can be recovered, or in determining the external information necessary for such recovery [59]. The key to this approach lies in leveraging the invariance encoded by conditional independencies within the causal graph.

Fairness and Missing data: Recent studies investigating the impact of imputation on algorithmic fairness under different missingness mechanisms reveal significant gaps. For instance, [92] presents theoretical results on fairness guarantees in the analysis of incomplete data, while [38] highlights common disparities in imputation quality across different demographic groups. Causal modeling has been pivotal in examining the relationship between fairness and the need to consider data missingness to achieve algorithmic fairness. In this vein, recent research has harnessed the power of causal modeling to unravel multivariate dependencies in datasets with missing data, exploring the sufficient and necessary conditions for recoverability of the distribution especially when multiple variables suffer from missing data [28, 59, 60]. In particular, [28] underscores that neglecting missing data can compromise the fairness of ML models, especially in high-stakes situations like loan decision-making. The authors of this study propose a novel algorithm with a decentralized decision-making process that only leverages recoverable conditional distributions when the joint data distribution is not recoverable.

3.1.2 Selection Bias

The sampling or selection of tuples in a dataset can be modeled through a selection variable C. This binary variable indicates whether a tuple is selected, i.e., the observed data can be viewed as a random sample from the distribution $\Pr(\boldsymbol{V} \mid C = 1)$, where $\boldsymbol{V}$ represents the set of all variables. In the case of a completely random selection mechanism, where C is independent of $\boldsymbol{V}$ (i.e., $C \perp\!\!\!\perp \boldsymbol{V}$), the sampled data distribution $\Pr(\boldsymbol{V} \mid C = 1)$ is representative of the underlying distribution $\Pr(\boldsymbol{V})$. However, in the presence of selection bias, where the selection process is non-random, the selection variable C becomes dependent on other variables (i.e., $C \not\perp\!\!\!\perp \boldsymbol{V}$). This dependency is depicted in the causal graph by edges from variables that affect the selection of data to the

variable C, capturing factors influencing data selection. As a result, the sampled data becomes biased and not representative of the underlying distribution, as indicated by $\Pr(\boldsymbol{V}) \neq \Pr(\boldsymbol{V} \mid C = 1)$.

Example 6 (Predictive Policing) *Figure 1b presents a simplified causal graph that captures the data collection process in predictive policing, where ML models are applied to predict crime. The graph encodes that crime risk Y is influenced by causal factors $\boldsymbol{W}$ such as education and income, but is independent of Race. However, the graph also highlights the bias in police data, which often reflects biases from individuals' interactions with the police, influenced by socio-cultural traits and patrol frequency in their neighborhoods [9, 53]. This reflects a case of non-random data selection, where the selection variable C is influenced by both the neighborhood (Zip) and socio-cultural traits ($\boldsymbol{Z}$), as depicted in Figure 1b. As a result, the police data can be viewed as a sample from $\Pr(\boldsymbol{V} \mid C = 1)$, where $\boldsymbol{V} = \{Race, \boldsymbol{Z}, Zip, \boldsymbol{X}, Y\}$ represents the set of all variables. Due to selection bias, conditioning on C introduces a spurious correlation between race and crime ($Race \not\perp\!\!\!\perp Y \mid C = 1$) in the training data, a phenomenon known as collider bias, which is depicted by bidirectional dotted red arrows between them in the graph. Training an ML model on this biased dataset to predict crime risk is likely to learn and propagate this spurious correlation, utilizing race in predicting crime, leading to unfair and inaccurate outcomes.*

Significant efforts in ML have been directed towards mitigating selection bias, employing various techniques including causal modeling to establish when it is fundamentally possible to recover from such biases [5, 6]. Within this scope, a prominent manifestation of selection bias is termed covariate shift, which occurs when there is a discrepancy in the distribution of features $\boldsymbol{X}$ between the training and test data, while the conditional distribution $\Pr(Y \mid \boldsymbol{X})$ remains constant. This phenomenon often arises when training data suffers from selection bias where the selection mechanism is independent of the label Y. This implies that the selection variable does not directly depend on the training label Y and is d-separated from it by $\boldsymbol{X}$ in the causal diagram.

Fairness and Selection bias: Recent work in ML has focused on the interaction between algorithmic fairness and selection bias [17, 20, 36, 52, 70]. These works, including inverse propensity scoring and density ratio estimation, often rely on specific assumptions about the underlying data distribution or the need for access to unbiased samples, a requirement that can be restrictive in practical scenarios. This challenge is particularly pronounced in sensitive areas such as predictive policing, healthcare, and finance, where inherent biases in these fields make obtaining unbiased data samples impossible. However, it is often more practical to acquire background knowledge about the data collection process in these domains. Such knowledge can be effectively represented through causal diagrams. In this vein, [78] introduces a method that uses causal diagrams to mitigate model unfairness, especially under covariate shift scenarios, although this method is applicable primarily to addressable graphs that satisfy certain graphical conditions.

To overcome the limitations encountered in previous methods, a recent study CRAB [96] introduces an approach for constructing fair ML models in the presence of selection bias, without the need for an unbiased dataset. Instead of relying on stringent assumptions or unbiased samples from the underlying distribution, CRAB only requires partial knowledge about the data collection process. This approach makes it more practical compared to other methodologies that necessitate more restrictive conditions. Next, we will review CRAB as a case study to illustrate how causal reasoning can be effectively utilized to develop ML models that maintain fairness in the underlying distribution, even when faced with selection bias.

3.1.3 Consistent Range Approximation for Building Fair Models under Selection Bias

CRAB presents a framework for developing fair models under selection bias, tailored to enforce fairness definitions that can be captured by conditional independence constraints, such as conditional statistical parity, equality of odds, and predictive parity [85]. Central to this framework is fairness queries, which assess the fairness of a classifier h, which will be reviewed next.

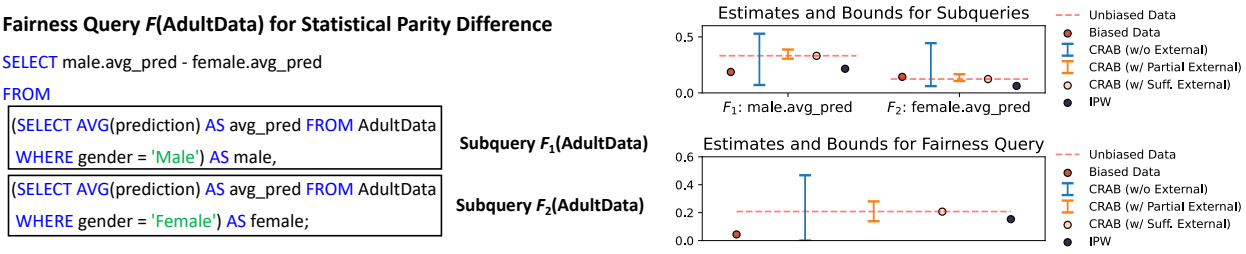


Figure 2: Comparative analysis of the consistent ranges obtained through CRA, alongside various estimates and the ground truth for the fairness query in the presence of selection bias in training data. In these plots, the red lines denote the fairness queries calculated using unbiased data. The fairness queries computed using biased data serve as a biased estimate of the unbiased fairness query. Another estimate of unbiased fairness query uses inverse propensity scores to re-weigh the data when evaluating fairness queries (IPW) [17]. The consistent ranges derived by CRAB with varying availability of external data are shown. Specifically, given sufficient external data, the consistent upper and lower bounds overlap [96].

Fairness query: Let h be a binary classifier with a protected attribute $S \in \boldsymbol{X}$, the fairness query is a measure used to assess the fairness violation of a model h wrt. the conditional statistical parity. It is defined based on a set of admissible attributes $\boldsymbol{A}$ and a population Ω with support $\boldsymbol{X} \times Y$:

$$F(\Omega) = \frac{1}{2\,|\boldsymbol{A}|} \sum_{\substack{y \in \textsc{Dom}(Y),\\ \boldsymbol{a} \in \textsc{Dom}(\boldsymbol{A})}} \left| \Pr_{\Omega}(h(\boldsymbol{x}) = y \mid s_1, \boldsymbol{a}) - \Pr_{\Omega}(h(\boldsymbol{x}) = y \mid s_0, \boldsymbol{a}) \right|.$$

It can be easily verified that for a model h to satisfy conditional statistical parity in a target population Ω, it must fulfill the condition $F(\Omega) = 0$. In this context, a fairness query is essentially the average dependency between the model's output and sensitive attributes, once adjustments have been made for admissible attributes. In practice, given a data D_Ω sampled from the distribution Ω, the fairness query $F(\Omega)$ can be computed through the empirical fairness query $\hat{F}(D_\Omega)$, which can be seen as an empirical estimate of $F(\Omega)$. Specifically, in the context of binary classification, $\hat{F}(D_\Omega)$ can be calculated by:

$$\hat{F}(D_\Omega) = \frac{1}{|\boldsymbol{A}|} \sum_{\boldsymbol{a} \in \textsc{Dom}(\boldsymbol{A})} \left| \frac{\sum_{\boldsymbol{x} \in \mathsf{N}^+_{s_1,\boldsymbol{a}}} h(\boldsymbol{x})}{|\mathsf{N}^+_{s_1,\boldsymbol{a}}|} - \frac{\sum_{\boldsymbol{x} \in \mathsf{N}^+_{s_0,\boldsymbol{a}}} h(\boldsymbol{x})}{|\mathsf{N}^+_{s_0,\boldsymbol{a}}|} \right|$$

where $\mathsf{N}^+_{s,\boldsymbol{a}}$ denotes the set of data points in D_Ω with positive labels, protected attribute values $S = s$ and admissible attributes value $\boldsymbol{A} = \boldsymbol{a}$. For example, Figure 2 presents an empirical fairness query that measures the model's violation of statistical parity on the adult data. In order to avoid sampling variability, in the subsequent, we assume samples are sufficiently large such that $\hat{F}(D_\Omega) \approx F(\Omega)$ and use them interchangeably.

Building models that are fair on the target population Ω requires to achieve $F(\Omega) = 0$. However, in practice, we only have access to the biased data D_Δ sampled from the population Δ that suffers from selection bias. Using this biased data D_Δ to evaluate fairness query gives $\hat{F}(D_\Delta)$, which is a biased and inaccurate estimate of the actual unfairness $F(\Omega)$. Furthermore, mitigating unfairness based on this biased estimate will result in a model that is fair on the biased training data ($F(\Delta) = 0$), while being unfair when deployed to the unbiased target population ($F(\Omega) \neq 0$). Nevertheless, without external data about the unbiased target population Ω, it's almost impossible to accurately estimate $F(\Omega)$.

In addressing the challenge of answering fairness queries from data affected by selection bias, the situation is akin to query answering on incomplete datasets, where complete and accurate responses are unattainable due to missing information. This challenge is tackled using an approach inspired by the concepts of possible worlds and consistent query answering [16, 24, 25, 47]. CRAB utilizes this methodology by considering every conceivable

underlying population or "possible world" from which the training data could have been obtained. Conceptually, CRAB computes the fairness query in each possible world and then uses these computations to establish a range for the fairness query by determining upper and lower bounds for unbiased fairness query answers. To generate tight and meaningful ranges, CRAB incorporates auxiliary information, which helps in narrowing down the range of potential underlying distributions. This approach is crucial for accurately evaluating fairness in models where the data is compromised by selection bias, ensuring a more reliable and valid assessment of fairness.

Auxiliary information and possible repairs: The auxiliary information that CRAB incorporates includes the causal diagram that represents the collection process of the biased data, and a set of external data sources that can potentially provide partial information about the underlying distribution Ω. Intuitively, the causal diagram encodes the causes of selection bias, i.e., which variables affect the tuple selection, while the external data source can be used to compute unbiased statistics about the underlying distribution. CRAB captures the space of possible unbiased, complete data using the notion of possible repairs. Formally, given a biased dataset D_Δ, the set of possible repairs of D_Δ, denoted as $\mathsf{Repairs}(D_\Delta)$, is defined as the set of all datasets D with the same schema as D_Δ such that: (1) $D \supsetneq D_\Delta$ and (2) D is consistent with $\mathcal{G}$ and $\mathcal{A}_\Omega$, i.e., it satisfies the constraints posed by the auxiliary information. Specifically, all repairs in $\mathsf{Repairs}(D_\Delta)$ must adhere to the conditional independences encoded in the causal diagram, and the unbiased statistics derived from $\mathcal{A}_\Omega$. Note that CRAB does not compute each of the possible repairs. Instead, the concept of possible repairs is used as a framework for addressing the incompleteness of information in the presence of selection bias. The problem of consistent range approximation is built upon the concept of possible repairs.

Consistent range approximation: The consistent range approximation (CRA) computes the consistent upper bound (CUB) and consistent lower bound (CLB) of the fairness query $F(\Omega)$. Similar to consistent query answering in databases [8, 19], CRA considers the space of all possible repairs, which stands for possible ways to complete the biased data D_Δ. Specifically,

$$\text{CLB} = \min_{D \in \mathsf{Repairs}(D_\Delta)} \hat{F}(D), \quad \text{CUB} = \max_{D \in \mathsf{Repairs}(D_\Delta)} \hat{F}(D)$$

As mentioned, CRA does not compute each of the possible repairs, but utilizes the conditional independence conditions encoded in the causal diagram, which every possible repair must satisfy, to derive closed-form solutions for the range of fairness query answers. This ensures that the actual unfairness of the model on the underlying distribution will fall within this computed range, i.e. $F(\Omega) \in [\text{CLB}, \text{CUB}]$. This range is referred to as the consistent range. Furthermore, CRA can integrate varying levels of external data sources about the underlying distribution, enabling the derivation of more precise consistent ranges. This property makes CRAB a practical solution for addressing selection bias.

It is worth noting that the external data source is not mandatory for CRA. In the absence of external data sources, [96] provides the closed-form CUB and CLB leveraging merely the conditional independence condition encoded in the causal diagram. We use the example of police data to demonstrate CRA in the absence of external data sources. For simplicity, we illustrate the CRA of fairness query wrt. statistical parity, where $\boldsymbol{A} = \emptyset$.

Example 7 (CRA on the Predictive Policing Data) *Continuing with Example 6, assume the protected attribute $Race \in \{white, non\text{-}white\}$ and the label, crime risk $Y \in \{low\ risk, high\ risk\}$. In this case, the fairness query wrt. statistical parity notion can computed by:*

$$F(\Omega) = \Pr_\Omega(low\ risk \mid white) - \Pr_\Omega(low\ risk \mid non\text{-}white). \tag{3}$$

The CUB of $F(\Omega)$ can be derived by combining the upper bound of $\Pr_\Omega(low\ risk \mid white)$ and the lower bound of $\Pr_\Omega(low\ risk \mid non\text{-}white)$. First, we show how $\Pr_\Omega(low\ risk \mid white)$ is upper bounded. As presented in Figure 1b, the selection variable C is influenced by ZIP and $\boldsymbol{Z}$. Let $\boldsymbol{U} = (C) = \{\boldsymbol{Z}, ZIP\}$, we have the conditional independence condition encoded in the causal diagram: $(C \perp\!\!\!\perp \boldsymbol{V} \mid \boldsymbol{U})$, where $\boldsymbol{V}$ is the set of all variables. The following holds due to this conditional independence:

$$\Pr_\Omega(low\ risk \mid white, \boldsymbol{u}) = \Pr_\Omega(low\ risk \mid white, \boldsymbol{u}, C = 1) = \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}). \tag{4}$$

The upper bound of $\Pr_\Omega(low\ risk \mid white)$ *can be derived by applying the law of total probability and Eq. 4:*

$$\begin{aligned}
\Pr_\Omega(low\ risk \mid white) &= \sum_{\boldsymbol{u} \in \text{DOM}(\boldsymbol{U})} \Pr_\Omega(low\ risk \mid white, \boldsymbol{u}) \Pr_\Omega(\boldsymbol{u} \mid white) \\
&= \sum_{\boldsymbol{u} \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}) \Pr_\Omega(\boldsymbol{u} \mid white) \\
&\leq \sum_{\boldsymbol{u} \in \text{DOM}(\boldsymbol{U})} (\max_{\boldsymbol{u}^* \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}^*))\ \Pr_\Omega(\boldsymbol{u} \mid white) \\
&= \max_{\boldsymbol{u}^* \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}^*) \sum_{\boldsymbol{u} \in \text{DOM}(\boldsymbol{U})} \Pr_\Omega(\boldsymbol{u} \mid white) \\
&= \max_{\boldsymbol{u}^* \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}^*).
\end{aligned} \tag{5}$$

Similarly, one can derive a lower bound for $\Pr_\Omega(low\ risk \mid non\text{-}white)$*, resulting in the subsequent formulation for the CUB of the fairness query:*

$$\mathit{F}(\Omega) \leq \mathit{CUB} = \max_{\boldsymbol{u}^* \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid white, \boldsymbol{u}^*) - \min_{\boldsymbol{u}^* \in \text{DOM}(\boldsymbol{U})} \Pr_\Delta(low\ risk \mid non\text{-}white, \boldsymbol{u}^*).$$

Furthermore, if sufficient external data sources which enable computing the unbiased statistics $\Pr_\Omega(\boldsymbol{u} \mid white)$ *are available, CRA is able to directly estimate* $\mathit{F}(\Omega)$*.*

The above results demonstrate how CRA gives consistent ranges with no or sufficient external data. In practice, one may have access to a level of external data that falls in between these two extremes. For instance, we might not have access to the external data about socio-cultrual traits $\boldsymbol{Z}$, thus only being able to compute the unbiased probabilities $\Pr_\Omega(ZIP \mid Race)$. CRAB also provides closed-form consistent ranges when having partial access to external data, including this case. Next, we empirically compare the various estimates of the fairness query with the CLBs and CUBs obtained through CRA on real-world data. We focus on the CLBs and CUBs computed when having no or sufficient external data, as they have been introduced in Example 7.

Example 8: Figure 2 presents the comparison between consistent ranges and the estimates of the model's unfairness on the unbiased distribution. The adult data [51], which contains financial and demographic data to predict if an individual's income exceeds $50K$, is used for model training and testing. Specifically, the training data is injected with selection bias, where the selection depends on gender, age, and relationship. In the example, the consistent range of the fairness query can be computed based on the consistent ranges of its sub-queries. When unbiased external data is unavailable, the fairness query computed using biased data shows significant inaccuracy, especially for subquery F_2. Nevertheless, in the absence of unbiased external data, CRAB guarantees to upper and lower bound the actual query answer on the underlying distribution. When the unbiased external data is leveraged, IPW still deviates from the unbiased fairness query. In contrast, given sufficient external data (a subset of unlabeled data used by IPW), the consistent upper and lower bounds derived by CRAB overlaps, resulting in an accurate estimate of the unbiased fairness query. In addition, the consistent ranges obtained with partial external data demonstrate the effectiveness of incorporating limited unbiased external data for deriving tighter consistent ranges. The results imply that (1) the consistent ranges always guarantee to bound the actual unfairness of the ML model, and (2) given external data about unbiased distribution, CRAB is able to derive tighter bounds or estimates of the unbiased fairness query.

The CUBs of fairness queries can be seen as the models' worst-case unfairness given available information about the underlying distribution. Therefore, CUBs can be used to train certifiably fair ML models by incorporating them into the loss function. In addition to the CRAB system, [96] also presents a theoretical analysis of the impact of selection bias on the fairness of ML models and establishes necessary and sufficient graphical conditions on the data collection causal diagram under which the selection bias leads to unfair ML models.

3.2 Confounding Bias

Confounding bias presents challenges in ML when a latent variable C confounds some observed features S with the training label Y, distorting their association. For example, in healthcare data, suppose S represents lifestyle factors or genetic predispositions, Y is the disease training label, and C encompasses unrecorded environmental factors like exposure to pollutants or access to healthcare facilities. Reliance on S for predicting Y can render ML models unreliable due to unstable correlations across different settings [91]. Furthermore, when S includes sensitive attributes, confounding bias can introduce biases that unfairly impact certain groups, especially if C relates to socioeconomic factors such as income level or education, thereby exacerbating disparities.

Example 9 (Medical Imaging) *Continuing with the application of skin cancer detection in Example 3. The causal modeling of confounding bias is shown in Figure 1c. In the causal diagram, the presence of a surgical skin marking (S) does not causally contribute to skin cancer (Y) as there are no edges between them. However, they become correlated in the data due to the confounding of disease severity (U).*

Since ML models learn correlation instead of causation, this non-causal spurious correlation between the presence of surgical skin markings and skin cancer will be learned and lead to inaccurate predictions. In particular, the model will have a high false positive rate on patients with other severe diseases, who are also likely to have surgical skin markings.

Fairness, Robustness, and Confounding Bias: Confounding bias poses a significant challenge across the board, particularly impacting the robustness and fairness of algorithmic models. The crux of efforts in algorithmic fairness is to ensure that sensitive attributes and training labels remain independent, conditioned on a subset of observed features, thus aiming to nullify spurious correlations brought about by unobserved confounding biases [26, 55, 74]. Achieving such independence ($S \perp\!\!\!\perp Y \mid X$), as exemplified in Example 9, is vital for preventing reliance on non-causal features like surgical markings for predictions, which enhances both the fairness and robustness of models. A variety of approaches have been developed to enforce conditional independence, ranging from feature selection methods that mitigate spurious correlations [26], counterfactual data augmentation techniques that elucidate causal relationships and generate varied counterfactual scenarios [55], to minimal repair strategies such as *Capuchin* for data adjustment in compliance with Multivalued Dependency (MVD) [74]. Furthermore, in-processing techniques play a crucial role, incorporating strategies such as integrating conditional mutual information into the loss function [77], employing adversarial mechanisms for confounding-invariant feature extraction [94], and developing feature representations that achieve conditional independence [83]. Ultimately, causal inference stands as a foundational strategy for modeling confounding bias and securing the requisite conditional independence, thus bolstering the efforts to enhance fairness and ensure robustness against confounding bias and spurious correlations in algorithmic models.

3.3 Measurement Bias

Given a variable U affected by the measurement error, we can create a variable U^* indicating the collected or observed values, while the actual variable U is unobserved. When measurement errors are non-random, the values of the observed variable U^* often depend on its actual value U and other variables. The observed data suffering from this measurement bias can be seen as a random sample from $\Pr(\boldsymbol{V} \setminus \{U\}, U^*)$ where $\boldsymbol{V}$ denotes the set of all variables. It is only representative of the underlying distribution when $\Pr(\boldsymbol{V}) = \Pr(\boldsymbol{V} \setminus \{U\}, U^*)$, which rarely holds in practice. In the context of ML, the label variable Y often suffers from mismeasurement and appears to be biased, which degrades the performance of downstream ML models [39]. Next, we will discuss an example of label bias existing in the medical imaging data.

Example 10 (Modeling Label Bias) *Continuing with the scenario of facial identification in Example 10. Figure 1d presents the causal modeling of label bias in this application. Ideally, the actual label Y and the sensitive attribute $Race$ are independent ($Y \perp\!\!\!\perp Race$). However, due to the inadvertent bias during the labeling, the observed label Y^* is influenced by both $Race$ and Y, resulting in the correlation between race-related facial features and labels in the observed data ($Y^* \not\perp\!\!\!\perp Race$). Consequently, models trained on this biased observed data will predict based on race-related facial features, leading to inaccuracies and unfairness.*

The general problem of measurement bias has been studied recently, particularly in the context of causal inference. Through structural equation modeling, [46] detects measurement bias in longitudinal health-related data. In contrast, [7] applies Bayesian factor analysis to effectively detect both uniform and non-uniform measurement bias, with high detection rates in cases where an observed violator is present. To eliminate the systematic bias induced by measurement errors, [65] highlights several algebraic and graphical methods that work under different assumptions about the error mechanism. Beyond the broader issue of measurement bias, a range of research specifically targets the challenge of unfairness stemming from label bias.

Fairness and Label Bias: Addressing label bias in ML necessitates innovative optimization and modeling strategies. [39] tackles this by positing that the distribution of biased labels should closely match the true distribution in terms of KL divergence, subject to the observed level of unfairness. They approach this through a constrained optimization problem, adjusting data weights to mitigate label bias while aiming for minimal alteration. [90] approaches fairness through a label-flipping optimization problem, designed to adjust labels for individual fairness with minimal changes, formulated as a mixed-integer quadratic programming problem. This is further refined to an integer linear programming challenge, with [90] providing approximate yet theoretically grounded solutions. On another front, [93] focuses on identifying label inaccuracies by associating low self-confidence in model predictions with potential errors, utilizing confidence intervals for selective data refinement. These methods, while effective, often rely on simplifying assumptions, such as a minimal number of mislabeled instances, and do not fully confront measurement bias directly. However, advancements in causal modeling offer a principled approach to constructing fair and accurate models by accounting for measurement bias. [14] leverage the concept of conditional independence between unbiased labels and other variables, informed by facial action units, to tailor loss functions that enhance fairness in facial expression recognition. Similarly, [18] explores various strategies for remedying label bias, emphasizing the crucial role of accurate causal diagrams in developing unbiased algorithmic risk assessments without compromising fairness.

4 Conclusions and Future Directions

This paper has investigated the significant challenges posed by data biases in machine learning (ML), emphasizing the critical role of causal modeling in addressing these complexities. By analyzing data biases resulting from missing data, confounding variables, and measurement errors, we have highlighted their substantial impact on the fairness, accuracy, and reliability of ML models. Adopting a causal perspective not only helps in mitigating the symptoms of data biases but also in directly tackling their root causes. This approach is key to developing more robust and equitable ML applications, illustrating the importance of understanding data generation processes to effectively minimize algorithmic bias.

Our exploration underscores the need for ongoing research and improvement in data-centric methods to enhance fairness, robustness, and accuracy in ML. We advocate for better data management practices, emphasizing their vital role in advancing ML and ensuring its benefits to society. Future research directions are poised for significant advances through the integration of data bias considerations with various aspects of data quality management in databases, particularly in terms of information incompleteness and inconsistency. Data biases inherently lead to these issues, suggesting that insights from data management research could significantly contribute to developing new approaches for data cleaning and quality management in ML. This includes devising strategies for training ML models in the presence of incomplete and uncertain data.

Moreover, effectively addressing data biases involves focusing on various constraints that capture the statistical properties of data, similar to integrity constraints in data management. Conditional independence constraints, for example, are a critical category of statistical integrity constraints vital for learning de-confounded predictive models, eliminating spurious correlations, and ensuring fairness in predictive modeling. The pursuit of research in developing data cleaning methods with respect to conditional independence constraints, investigating the interplay between these constraints and database dependencies, and formulating efficient maintenance, validation, and repair techniques is imperative. Such initiatives are poised to significantly enhance data fairness and model reliability in ML, paving the way for more accountable and transparent AI systems.

References

[1] Karen Antman, David Amato, William Wood, J Carson, Herman Suit, Karl Proppe, Robert Carey, J Greenberger, R Wilson, and E Frei 3rd. Selection bias in clinical trials. Journal of Clinical Oncology, 3(8):1142–1147, 1985.

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. arXiv preprint arXiv:1907.02893, 2019.

[3] Agathe Balayn, Christoph Lofi, and Geert-Jan Houben. Managing bias and unfairness in data for decision support: a survey of machine learning and data engineering approaches to identify and mitigate bias and unfairness within data management and analytics systems. The VLDB Journal, 30(5):739–768, 2021.

[4] John Banasik, Jonathan Crook, and Lyn Thomas. Sample selection bias in credit scoring models. Journal of the Operational Research Society, 54(8):822–832, 2003.

[5] Elias Bareinboim and Judea Pearl. Controlling selection bias in causal inference. In Artificial Intelligence and Statistics, pages 100–108. PMLR, 2012.

[6] Elias Bareinboim, Jin Tian, and Judea Pearl. Recovering from selection bias in causal and statistical inference. In Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014.

[7] M. Barendse, C. Albers, F. Oort, and M. Timmerman. Measurement bias detection through bayesian factor analysis. Frontiers in Psychology, 5, 2014.

[8] Leopoldo Bertossi. Consistent query answering in databases. ACM Sigmod Record, 35(2):68–76, 2006.

[9] Sarah Brayne, Alex Rosenblat, and Danah Boyd. Predictive policing. Data & Civil Rights: A New Era Of Policing And Justice, pages 2015–1027, 2015.

[10] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. Data Mining and Knowledge Discovery, 21(2):277–292, 2010.

[11] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. Advances in neural information processing systems, 30, 2017.

[12] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In Proceedings of the conference on fairness, accountability, and transparency, pages 319–328, 2019.

[13] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. Bias in machine learning software: Why? how? what to do? arXiv preprint arXiv:2105.12195, 2021.

[14] Yunliang Chen and Jungseock Joo. Understanding and mitigating annotation bias in facial expression recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 14980–14991, 2021.

[15] SC Choi and IL Lu. Effect of non-random missing data mechanisms in clinical trials. Statistics in medicine, 14(24):2675–2684, 1995.

[16] Marco Console, Paolo Guagliardo, Leonid Libkin, and Etienne Toussaint. Coping with incomplete data: Recent advances. In Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, pages 33–47, 2020.

[17] Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In International conference on algorithmic learning theory, pages 38–53. Springer, 2008.

[18] Jessica Dai and Sarah M Brown. Label bias, label shift: Fair machine learning with unreliable labels. In NeurIPS 2020 Workshop on Consequential Decision Making in Dynamic Environments, volume 12, 2020.

[19] Akhil A Dixit and Phokion G Kolaitis. Consistent answers of aggregation queries using sat solvers. arXiv preprint arXiv:2103.03314, 2021.

[20] Wei Du and Xintao Wu. Robust fairness-aware learning under sample selection bias. arXiv preprint arXiv:2105.11570, 2021.

[21] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. In ITCS, pages 214–226. ACM, 2012.

[22] Adrien Ehrhardt, Christophe Biernacki, Vincent Vandewalle, Philippe Heinrich, and Sébastien Beben. Reject inference methods in credit scoring. Journal of Applied Statistics, 48(13-15):2734–2754, 2021.

[23] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. nature, 542(7639):115–118, 2017.

[24] Wenfei Fan and Floris Geerts. Foundations of data quality management. Synthesis Lectures on Data Management, 4(5):1–217, 2012.

[25] Su Feng, Boris Glavic, Aaron Huber, and Oliver A Kennedy. Efficient uncertainty tracking for complex queries with attribute-level bounds. In Proceedings of the 2021 International Conference on Management of Data, pages 528–540, 2021.

[26] Sainyam Galhotra, Karthikeyan Shanmugam, Prasanna Sattigeri, and Kush R Varshney. Causal feature selection for algorithmic fairness. 2022.

[27] Milena A Gianfrancesco, Suzanne Tamang, Jinoos Yazdany, and Gabriela Schmajuk. Potential biases in machine learning algorithms using electronic health record data. JAMA internal medicine, 178(11):1544–1547, 2018.

[28] Naman Goel, Alfonso Amayuelas, Amit Deshpande, and Amit Sharma. The importance of modeling data missingness in algorithmic fairness: A causal perspective. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pages 7564–7573, 2021.

[29] John Graham. Missing data: Analysis and design. New York, NY: Springer. 06 2012.

[30] Gareth J Griffith, Tim T Morris, Matthew J Tudball, Annie Herbert, Giulia Mancano, Lindsey Pike, Gemma C Sharp, Jonathan Sterne, Tom M Palmer, George Davey Smith, et al. Collider bias undermines our understanding of covid-19 disease risk and severity. Nature communications, 11(1):1–12, 2020.

[31] Shubha Guha, Falaah Arif Khan, Julia Stoyanovich, and Sebastian Schelter. Automated data cleaning can hurt fairness in machine learning-based decision making. ICDE, 2022.

[32] Anna Guo, Jiwei Zhao, and Razieh Nabi. Sufficient identification conditions and semiparametric estimation under missing not at random mechanisms. arXiv preprint arXiv:2306.06443, 2023.

[33] Luke Haliburton, Sinksar Ghebremedhin, Robin Welsch, Albrecht Schmidt, and Sven Mayer. Investigating labeler bias in face annotation for machine learning. arXiv preprint arXiv:2301.09902, 2023.

[34] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In NIPS, pages 3315–3323, 2016.

[35] Ziniu Hu, Zhe Zhao, Xinyang Yi, Tiansheng Yao, Lichan Hong, Yizhou Sun, and Ed Chi. Improving multi-task generalization via regularizing spurious correlation. Advances in Neural Information Processing Systems, 35:11450–11466, 2022.

[36] Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex Smola. Correcting sample selection bias by unlabeled data. Advances in neural information processing systems, 19, 2006.

[37] Maliha Tashfia Islam, Anna Fariha, Alexandra Meliou, and Babak Salimi. Through the data management lens: Experimental analysis and evaluation of fair classification. In Proceedings of the 2022 International Conference on Management of Data, pages 232–246, 2022.

[38] Vincent Jeanselme, Maria De-Arteaga, Zhe Zhang, Jessica Barrett, and Brian Tom. Imputation strategies under clinical presence: Impact on algorithmic fairness. In Machine Learning for Health, pages 12–34. PMLR, 2022.

[39] Heinrich Jiang and Ofir Nachum. Identifying and correcting label bias in machine learning. In International Conference on Artificial Intelligence and Statistics, pages 702–712. PMLR, 2020.

[40] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. Knowledge and information systems, 33(1):1–33, 2012.

[41] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In 2012 IEEE 12th International Conference on Data Mining, pages 924–929. IEEE, 2012.

[42] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 35–50. Springer, 2012.

[43] Bojan Karlas, Peng Li, Renzhi Wu, Nezihe Merve Gürel, Xu Chu, Wentao Wu, and Ce Zhang. Nearest neighbor classifiers over incomplete information: From certain answers to certain predictions. Proc. VLDB Endow., 14(3):255–267, 2020.

[44] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In Proceedings of the 35th International Conference on Machine Learning, pages 2564–2572. PMLR, 2018.

[45] Jae Kwang Kim. Finite sample properties of multiple imputation estimators. 2004.

[46] B. King-Kallimanis, F. Oort, and G. Garst. Using structural equation modelling to detect measurement bias and response shift in longitudinal data. AStA Advances in Statistical Analysis, 94:139–156, 2010.

[47] Paraschos Koutris and Jef Wijsen. Consistent query answering for primary keys and conjunctive queries with negated atoms. In Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, pages 209–224, 2018.

[48] Sanjay Krishnan, Jiannan Wang, Eugene Wu, Michael J Franklin, and Ken Goldberg. Activeclean: Interactive data cleaning for statistical modeling. Proceedings of the VLDB Endowment, 9(12):948–959, 2016.

[49] Manabu Kuroki and Judea Pearl. Measurement bias and effect restoration in causal inference. Biometrika, 101(2):423–437, 2014.

[50] Katherine J Lee and John B Carlin. Recovery of information from multiple imputation: a simulation study. Emerging themes in epidemiology, 9:1–10, 2012.

[51] M. Lichman. Uci machine learning repository, 2013.

[52] Anqi Liu and Brian Ziebart. Robust classification under sample selection bias. Advances in neural information processing systems, 27, 2014.

[53] Kristian Lum and William Isaac. To predict and serve? Significance, 13(5):14–19, 2016.

[54] Qingwei Luo, Sam Egger, Xue Qin Yu, David P Smith, and Dianne L O'Connell. Validity of using multiple imputation for" unknown" stage at diagnosis in population-based cancer registry data. PLoS One, 12(6):e0180033, 2017.

[55] Jing Ma, Ruocheng Guo, Aidong Zhang, and Jundong Li. Learning for counterfactual fairness from observational data. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 1620–1630, 2023.

[56] Mohammad Mahdavi and Ziawasch Abedjan. Baran: Effective error correction via a unified context representation and transfer learning. Proceedings of the VLDB Endowment, 13(12):1948–1961, 2020.

[57] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. ACM Computing Surveys (CSUR), 54(6):1–35, 2021.

[58] Roger E Millsap and Howard T Everson. Methodology review: Statistical approaches for assessing measurement bias. Applied psychological measurement, 17(4):297–334, 1993.

[59] Karthika Mohan and Judea Pearl. Graphical models for processing missing data. Journal of the American Statistical Association, 116(534):1023–1037, 2021.

[60] Razieh Nabi, Rohit Bhattacharya, Ilya Shpitser, and James Robins. Causal and counterfactual views of missing data models. arXiv preprint arXiv:2210.05558, 2022.

[61] Felix Neutatz, Binger Chen, Ziawasch Abedjan, and Eugene Wu. From cleaning before ml to cleaning for ml. Data Engineering, page 24, 2021.

[62] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. Social data: Biases, methodological pitfalls, and ethical boundaries. Frontiers in Big Data, 2:13, 2019.

[63] L. Page and M. Henderson. Appraising the evidence: what is measurement bias? Evidence Based Mental Health, 11:36 – 37, 2008.

[64] Judea Pearl. Causality. Cambridge university press, 2009.

[65] Judea Pearl. On measurement bias in causal inference. arXiv preprint arXiv:1203.3504, 2012.

[66] Judea Pearl and Karthika Mohan. Recoverability and testability of missing data: Introduction and summary of results. Available at SSRN 2343873, 2013.

[67] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. Advances in neural information processing systems, 30, 2017.

[68] Jennifer K Plichta, Christel N Rushing, Holly C Lewis, Marguerite M Rooney, Dan G Blazer, Samantha M Thomas, E Shelley Hwang, and Rachel A Greenup. Implications of missing data on reported breast cancer mortality. Breast Cancer Research and Treatment, 197(1):177–187, 2023.

[69] Theodoros Rekatsinas, Xu Chu, Ihab F. Ilyas, and Christopher Ré. Holoclean: Holistic data repairs with probabilistic inference. Proc. VLDB Endow., 10(11):1190–1201, 2017.

[70] Ashkan Rezaei, Anqi Liu, Omid Memarrast, and Brian Ziebart. Robust fairness under covariate shift. arXiv preprint arXiv:2010.05166, 2020.

[71] A. Rosenberg, V. Dussel, L. Orellana, T. Kang, J. Geyer, C. Feudtner, and J. Wolfe. What's missing in missing data? omissions in survey responses among parents of children with advanced cancer. Journal of palliative medicine, 17 8:953–6, 2014.

[72] Donald B Rubin. Multiple imputations in sample surveys-a phenomenological bayesian approach to nonresponse. In Proceedings of the survey research methods section of the American Statistical Association, volume 1, pages 20–34. American Statistical Association Alexandria, VA, USA, 1978.

[73] Babak Salimi, Bill Howe, and Dan Suciu. Database repair meets algorithmic fairness. ACM SIGMOD Record, 49(1):34–41, 2020.

[74] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. Interventional fairness: Causal database repair for algorithmic fairness. In Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019., pages 793–810, 2019.

[75] Sebastian Schelter, Tammo Rukat, and Felix Biessmann. Jenga-a framework to study the impact of data errors on the predictions of machine learning models. In EDBT, pages 529–534, 2021.

[76] Nathaniel Schenker and Alan H Welsh. Asymptotic results for multiple imputation. The Annals of Statistics, 16(4):1550–1566, 1988.

[77] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Information-theoretic bias reduction via causal view of spurious correlation. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, pages 2180–2188, 2022.

[78] Harvineet Singh, Rina Singh, Vishwali Mhasawade, and Rumi Chunara. Fairness violations and mitigation under covariate shift. In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, pages 3–13, 2021.

[79] Anita Johanna Tørmoen, Martin Øverlien Myhre, Anine Therese Kildahl, Fredrik Andreas Walby, and Ingeborg Rossow. A nationwide study on time spent on social media and self-harm among adolescents. Scientific reports, 13(1):19111, 2023.

[80] Geert Verstraeten and Dirk Van den Poel. The impact of sample bias on consumer credit scoring performance and profitability. Journal of the operational research society, 56:981–992, 2005.

[81] Julia K Winkler, Christine Fink, Ferdinand Toberer, Alexander Enk, Teresa Deinlein, Rainer Hofmann-Wellenhof, Luc Thomas, Aimilios Lallas, Andreas Blum, Wilhelm Stolz, et al. Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition. JAMA dermatology, 155(10):1135–1141, 2019.

[82] Blake Woodworth, Suriya Gunasekar, Mesrob I. Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In Proceedings of the 2017 Conference on Learning Theory, pages 1920–1953, 2017.

[83] Renzhe Xu, Peng Cui, Kun Kuang, Bo Li, Linjun Zhou, Zheyan Shen, and Wei Cui. Algorithmic decision making with conditional fairness. In Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, pages 2125–2135, 2020.

[84] Tal Yarkoni. The generalizability crisis. Behavioral and Brain Sciences, 45:e1, 2022.

[85] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In Proceedings of the 26th International Conference on World Wide Web, pages 1171–1180. International World Wide Web Conferences Steering Committee, 2017.

[86] John R Zech, Marcus A Badgeley, Manway Liu, Anthony B Costa, Joseph J Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. PLoS medicine, 15(11):e1002683, 2018.

[87] Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. In ICML (3), volume 28 of JMLR Workshop and Conference Proceedings, pages 325–333. JMLR.org, 2013.

[88] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, pages 335–340, 2018.

[89] Hantian Zhang, Xu Chu, Abolfazl Asudeh, and Shamkant B Navathe. Omnifair: A declarative system for model-agnostic group fairness in machine learning. In Proceedings of the 2021 international conference on management of data, pages 2076–2088, 2021.

[90] Hantian Zhang, Ki Hyun Tae, Jaeyoung Park, Xu Chu, and Steven Euijong Whang. iflipper: Label flipping for individual fairness. Proceedings of the ACM on Management of Data, 1(1):1–26, 2023.

[91] Xingxuan Zhang, Peng Cui, Renzhe Xu, Linjun Zhou, Yue He, and Zheyan Shen. Deep stable learning for out-of-distribution generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5372–5382, 2021.

[92] Yiliang Zhang and Qi Long. Assessing fairness in the presence of missing data. Advances in Neural Information Processing Systems, 34, 2021.

[93] Yixuan Zhang, Boyu Li, Zenan Ling, and Feng Zhou. Mitigating label bias in machine learning: Fairness through confident learning. arXiv preprint arXiv:2312.08749, 2023.

[94] Qingyu Zhao, Ehsan Adeli, and Kilian M Pohl. Training confounder-free deep learning models for medical applications. Nature communications, 11(1):1–9, 2020.

[95] Sami Zhioua and Rūta Binkytė. Dissecting causal biases. 2023.

[96] Jiongli Zhu, Sainyam Galhotra, Nazanin Sabri, and Babak Salimi. Consistent range approximation for fair predictive modeling. Proceedings of the VLDB Endowment, 16(11):2925–2938, 2023.

Fairness and Robustness in Answering Preference Queries

Senjuti Basu Roy
New Jersey Institute of Technology
senjutib@njit.edu

Abstract

Given a large number of users preferences as inputs over a large number of items, preference queries leverage different preference aggregation methods to aggregate individual preferences in a systematic manner and come up with a single output (top-k ordered or unordered/a complete order) that is most representative. The preference aggregation methods are widely adopted from the social choice theory, some of which are rank based (single-round vs. multi-round), while others are non-rank based. These queries are prevalent in high fidelity applications, including search, ranking and recommendation, hiring and admission, and electoral voting systems. This article outlines algorithmic challenges and directions in designing an optimization guided computational framework that allows to change the original aggregated output (either ordered or unordered top-k or a complete order) to satisfy different criteria related to fairness and robustness, considering different preference elicitation models (ways users provide their input preferences) and aggregation methods (ways the individual preference get aggregated).

1 Introduction

The need to aggregate a large number of individual preferences in a systematic manner is ubiquitous. Users can provide preferences in many ways - as likes/dislikes, ordinal preferences, or ranked order (full or partial). The social choice theory [17] offers a plethora of aggregation methods to aggregate individual preferences and come up with a single output. These outputs may be a single rank that is most representative of all users preferences, or sometimes a smaller number of k items (top-k) that are ordered or presented as a set. While designed for electoral voting systems primarily, the applicability of answering queries is prevalent in many high fidelity applications, such as, ranking and listing web search results, recommending movies/songs, selecting a handful of candidates for domains where resource is scarce (such as hiring and admission), to name a few. It is not a stretch to consider a setting in which thousands of items (notationally n) have received preferences from hundreds of thousands (or even millions) of users (notationally m) and the goal is to produce a single output (notationally σ) that is most representative.

The computational implications of different preference aggregation methods are well studied. What is not so well understood is how hard it is to change the original produced output, which may be necessary for many compelling reasons. Satisfying additional criteria, such as, promoting fairness (e.g., ensuring presence of individuals with certain socio-demographic properties), or Understanding robustness, i.e., figuring out the minimum amount of change of the inputs that would result in a different outcome than the original output. This latter aspect provides understanding on how manipulable the proposed aggregation methods are which

Copyright 2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

are certainly important for aggregation methods that are heavily used in electoral systems, but are applicable in other scenarios as well (e.g., in figuring out the robustness of a rating system of products). To the best of our knowledge, a systematic study is needed to investigate these aspects in conjunction with different preference elicitation models, requiring different preference aggregation methods. That, in nutshell, is the focus of this article.

We discuss these challenges considering four interspersed dimensions, as described below.

Preference Elicitation Models. The article simultaneously considers a vast range of preference elicitation processes that we broadly categorize as rank based and non rank based. In rank based processes, the users can provide a fully ranked order over all items, a partial order, or a coarser preference (like item a ranked higher than item b, etc). In non rank based preferences, users can provide only likes, both likes and dislikes, or even an ordinal preference (likes item a as "excellent", b as "good", etc). The choice of these preference elicitation methods is dictated by the different applications. Rank based ones are suitable in hiring/admission/electoral system, while non rank based ones are more relevant in obtaining user feedback from search results, user satisfaction survey, product reviews, etc.

Preference Aggregation Methods. Then the preference aggregation methods that are most commensurate to the underlying preference elicitation process and underlying application are studied. For example, when user preferences are given as ranked order, depending on the underlying application, we will aggregate them using existing single-round rank based methods (e.g., Kemeny, Spearman's footrule, or Borda), or multi-round based methods (STV, IRV). The former aggregation methods are suitable in hiring decision, whereas, the latter ones are gaining popularity in voting systems. On the other hand, when users provide non rank based preferences, we will show how Jaccard similarity or Hamming distances are suitable to aggregate them and come up with the final output.

Produced Output Form. From the application point of view, the produced output may require an order over all n items (hiring/admission), or a small number k of n items as outputs. In case of top-k items requirement, the returned k-items may need to be ordered for certain applications (top-k web pages returned by the search engine), or in some cases it is fine to return them as a set (selecting a set of representatives or body to form certain committee).

Change Original Output. The importance of quantifying the minimum effort needed to change the original output is evident for several reasons, such as promoting fairness and robustness. Robustness is heavily used in electoral system to produce margin, that investigates how to bound the amount of change of the original outcome in case x% of the inputs are destroyed/deleted/modified. We discuss them further in details below.

2 Overarching Research Goals

The overarching goal is to design optimization guided computational framework containing principled models and scalable solutions that allows to change the original aggregated output (either ordered or unordered top-k or a complete order) to satisfy different criteria, considering different preference elicitation models and aggregation functions to promote: **a. Fairness** from the standpoint of the protected attributes[27] of the items/candidates (e.g., race, gender, ethnicity), where the candidates are selected by aggregating elicited preferences of the members (panelists, voters, search committee). We shall investigate existing group fairness criteria in the context of preference aggregation [27, 29], as well as adapt fairness criteria studied in the context of resource allocation or social choice theory. **b. Robustness**, namely, understanding how easy or hard it is to change the original outcome of different preference aggregation models given a budgeted preference substitution requirement. For instance, if the total number of preference updates is budgeted to be $\leq x$, is it possible to change the original outcome? We are interested in exploring these viewpoints for multiple preference elicitation models and output forms. What is also important to notice is that a given preference elicitation may be suitable to multiple aggregation methods and may require to satisfy more than one produced output form. These gives rise to many combinations of the

problem.

The rest of the article is organized as follows: In Section 4, we study how to Satisfy output constraints in single round rank-based preference aggregation methods. We study this considering ranking, which is a commonly used method to prioritize desirable outcomes among a set of items/candidates and is an essential step in many high impact applications. Here the members elicit a complete or partial preference order over the candidates and the goal is to produce an aggregated ranked order over all candidates or produce top-k results that minimize disagreements among individual preferences. We will also include preference substitution in single round rank-based preference aggregation methods to satisfy complex top-k constraints, where the requirement is defined over a set R of protected attributes.

In Section 5, we study how to satisfy output constraints in multi round rank-based preference aggregation methods, popularly known as ranked choice voting or (RCV) [12]. Two popular representatives of these models are IRV (Instant run-off voting) [12] that selects one item/candidate as the winner, and STV (single transferable vote) [10, 13] that generalizes IRV and selects a set of k-items/candidates as winners. It is known that RCV represents majority rules and improves result diversity. Unlike single round preference aggregation models, RCV minimizes the effect of strategic voting as users can provide their "true preference" for the candidates they support, not just provide preference against the items/candidates they oppose most. It is also shown in recent works, how RCV promotes anonymity and anti-plurality [13], compared to single round based algorithms.

In Section 6, we will study how to satisfy output constraints in non rank based preference aggregation methods. Here we investigate preference aggregation methods that do not require users inputs to be ranked order. A simple case in this context is a Boolean model, where each user describes their preference over n items as a Boolean vector of 0 and 1. When users provide only their "likes" on the items, the aggregation function such as Jaccard Similarity or Overlap similarity [24] may be appropriate to find top-k items that have exhibited maximum similarity over the users preferences. On the other hand, when the users provide both "likes" and "dislikes", the aggregation function may intend to produce a Boolean vector that minimizes the Hamming Distance between the input preferences and the produced output. Generalization of the Boolean preference elicitation models is also discussed.

Comparison with Existing Work. This contribution builds on our work recent works on fairness [20, 28], and prior works on preference aggregation [2, 3, 8], studying robustness [24]. We acknowledge that the existing popular group based fairness definition, such as, statistical parity [15] is somewhat similar to one of our proposed fairness notion. However, the best adapted version of top-k statistical parity studied in a recent paper [21] does not account for proportionate representation in every position of the top-k, limiting its applicability. Studying computational challenges related to computing the margin of victory has been a focus of recent research [4, 6, 10] in the context of electoral voting and related applications. But none of these existing works study the general version of the problem, which is, how to promote additional simple/complex constraints/criteria in the output, which is our primary focus. Other than these prior works, which are much narrow in scope, we are unaware of any computational work that systematically studies different preference elicitation models, multiple output changing criteria, and preference aggregation combining these two.

3 Formalism

There are 4 types of inputs that our proposed framework takes: (a) a set N of n items, where each item has a set $\mathcal{A}$ of discrete attributes. Each attribute $a \in \mathcal{A}$ has ℓ_a different values. (b) a set of m users, where the i-th user $u(i)$ provides her preference as σ_i. The users' preferences could be rank based, partial or full order, or non rank based. (c) a distance function $\mathcal{F}$ (defined formally below) that measure the "distance" between a set of m input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ and an output σ with the required output form. The exact distance function depends on the underlying preference elicitation model and the required output form which may be either a complete ranking of the items or a subset of k items, either ranked or not. (d) a set $\mathcal{C}$ of output criteria/constraints. Some variants of our problem also include as input a budgetary constraint B.

Definition 3.1: Distance function $\mathcal{F}$. Given m input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ and an output σ with the required output form, the function $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m)$ is the distance of σ from the input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$. In some cases the function $\mathcal{F}(\cdot)$ is an aggregation of a distance function between a single input preference and the output. Examples for such an aggregation are the sum of the pairwise distances and the maximum distance to any of the input preferences. In other cases $\mathcal{F}(\cdot)$ measures the minimum modification of the input preferences that would result in the preference aggregation method outputting the output σ.

Definition 3.2: Output criteria/constraints. For an attribute $a \in \mathcal{A}$, let $c(p_a)$ denote the cardinality constraints of items with value p_a (p_a is one of the ℓ_a possible values of attribute a). Given to the framework is a set C of such cardinality constraints for each attribute value p_a, for every $a \in A$, $A \subset \mathcal{A}$. There are two explicit cases that we consider.

- **The output σ is ordered and consists of $k \leq n$ items.** In this case the cardinality constraints are defined for every $\kappa \in [1..k]$ items, and for every such $\kappa \in [1..k]$, the κ top ranked items of output σ have to satisfy these cardinality constraints.
- **The output σ is an unordered set of k items.** In this case the cardinality constraints are defined for k items and the items in the output set σ have to satisfy these cardinality constraints.

Definition 3.3: A budgetary constraint. A budgetary constraint B is an upper bound on the distance of the output from the input preferences. The budgetary constraint implies that $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m) \leq B$.

Definition 3.4: Preference Aggregation Considering Constraints. We intend to study different types of problem definitions that require different algorithmic treatments. Given either complete or partial preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ over the items in N, a preference aggregation method, a distance function $\mathcal{F}(\cdot)$, and a set of output criteria $\mathcal{C}$.

- **(Constrained optimization).** Produce an output σ with the required form that minimizes $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m)$ and satisfies $\mathcal{C}$.
- **(Optimization under budgetary constraints).** Produce an output σ with the required form that optimizes $\mathcal{C}$, while satisfying $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m) \leq B$. (The objective function for optimizing $\mathcal{C}$ varies.)
- **(Bi-criteria optimization).** Given parameters α and β produce an output σ with the required form that satisfies both $\mathcal{F}(\sigma, \sigma_1, \ldots, \sigma_m) \leq \alpha$ and $\mathcal{G}(\mathcal{C}) \leq \beta$, where $\mathcal{G}$ is the objective function for optimizing $\mathcal{C}$.

3.1 Specifying Output Criteria

We discuss orthogonal reasons where the original outputs coming out of the preference aggregation methods need to be "massaged" further. What unifies them is that these criteria are defined over one or more attributes of the items. Depending on how many attributes are involved in the definition and their relationship thereof gives rise to additional challenges.

3.1.1 Fair Preference Aggregation

We will study fairness in the context of group based protected attributes of the candidates. Output criteria/constraints for fairness (refer to Definition 3.2) are expressed over one or more protected attributes. Their protected attributes could be expressed over gender, ethnicity, race, or the state the candidates are living in.

Formally speaking, each item/candidate $v \in N$ has one or more protected attributes. When $\ell_a = 2$, it is a binary protected attribute; when $\ell_a \geq 2$ it is a multi-valued protected attribute. As an example, race is (usually) a multi-valued protected attribute, and gender is sometimes a binary protected attribute.

p-fairness. p-fairness has been studied in the context of resource allocation satisfying temporal fairness or proportionate progress [7, 25]. It was introduced in the classical Chairman Assignment Problem [5, 25] that studies how to select a chairman of an union every year from a set of n states such that that at any time the accumulated number of chairmen from each state is proportional to its weight.

In the context of ranking, suppose that each of the n ranked items has a protected attribute $a(\cdot)$ that can take any of ℓ_a different values. For $p_a \in [1..\ell_a]$, let $c'(p_a)$ denote the fraction of items with protected attribute value p_a, that is, $c'(p_a) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{a(i)=p_a}$. The goal is to ensure that $c'(p_a)$ fraction (rounded either up or down) of every top κ items have protected attribute value p_a.

3.1.2 Robust Preference Aggregation

Output criteria/constraints for robustness on the other hand investigates the flip questions: Given either complete or partial preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ over n items, let σ be the output obtained by the preference aggregation method. Given a budget B, how to make B or less changes in the original preferences, such that the outcome is different from σ? This question is related to finding the margin in electoral systems and quantifies how manipulable the underlying aggregation method is. We study this problem under different manipulation models – addition only, deletion only, or substitution (addition + deletion).

4 Single Round Rank based Preference Aggregation

We outline two separate lines of algorithmic problems: (1) incorporating output criteria (e.g., p-fairness) in single round rank-based preference aggregation methods, and (2) satisfying complex constraints in single round rank-based preference aggregation methods.

4.1 Incorporating output criteria in rank aggregation

The input to the classical rank aggregation problem consists of m complete order of preferences over the n items/candidates. Traditionally, producing the final ranking involves aggregating potentially conflicting preferences from multiple individuals, and is known as the rank aggregation problem [1, 16, 26]. Our goal is to minimally change the aggregated output to enable fairness. We will study p-fairness [7, 25] that ensures proportionate representation of every group based on a protected attribute in every position of the aggregated ranked order. The classical problem in this context is known as the Chairman Assignment Problem [5, 25] which studies how to select a chairman of a union every year from a set of r states such that that at any time the accumulated number of chairmen from each state is proportional to its weight. p-fairness generalizes other notions of fairness [19] that were considered in prior work, including the existing popular group based fairness definition statistical parity [15].

4.1.1 Research Directions

Consider rankings of the items in a set V. Each such ranking can be viewed as a permutation. We will use the terms ranking and permutation interchangeably.

Kendall-Tau and Kemeny distances. Given two rankings $\sigma, \eta : V \to [1..n]$, the Kendall-Tau distance between the two rankings is the sum of pairwise disagreements between σ and η (bubble-sort distance)

$$\mathcal{K}(\sigma, \eta) = \sum_{\{u,v\} \subseteq V} \mathbb{1}_{(\sigma(v)-\sigma(u))(\eta(v)-\eta(u))<0}.$$

For a set of rankings $\{\eta_1, \eta_2, \ldots, \eta_m\}$ the Kemeny distance of the ranking σ to this set as

$$\kappa(\sigma, \eta_1, \eta_2, \ldots, \eta_m) = \sum_{i=1}^{m} \mathcal{K}(\sigma, \eta_i).$$

Spearman's footrule distance. Given two rankings $\sigma, \eta : V \to [1..n]$, the Spearman's footrule distance between the two rankings is the sum of the absolute values (ℓ_1 distance) of the differences between rankings σ and η.

$$\mathcal{S}(\sigma, \eta) = \sum_{u \in V} |(\sigma(u) - \eta(u)|$$

For a set of rankings $\{\eta_1, \eta_2, \ldots, \eta_m\}$ the Spearman's footrule distance of the ranking σ to this set is the sum of the pairwise distances.

Rank aggregation. The aggregated ranking of a set of m rankings $\{\rho_1, \rho_2, \ldots, \rho_m\}$ for a given distance function is a ranking that minimizes the distance to this set.

p-fairness for a ranking. For a permutation $\sigma, k \in [1..n], p \in [1..\ell]$, let $P(\sigma, k, p)$ denote the number of elements with protected attribute value p among the k top ranked elements in σ. A ranking σ is proportionate fair or p-fair if

$$\forall k \in [1..n] \, \forall p \in [1..\ell] : \; P(\sigma, k, p) \in \{\lfloor f(p) \cdot k \rfloor, \lceil f(p) \cdot k \rceil\}.$$

We formalized two optimization problems, individual p-fairness or **IPF** and the rank aggregation problem subject to proportionate fairness (**RAPF**) considering binary ($\ell = 2$) and multi-valued ($\ell > 2$) protected attributes. These problems and associated algorithmic results could be found in [28].

4.1.2 Open Problems

We plan to investigate the following open problems.

p-fairest aggregate ranking (PFAR). The PFAR problem is defined as follows. Given a set of m rankings choose the "p-fairest" ranking among all rankings that minimize the Kemeny distance to this set. We need to define "p-fairest" ranking or a distance measure to a p-fair ranking. We propose the following distance measure (using the notations defined above). For an integer $d \geq 0$, a ranking σ is at distance d from a p-fair ranking if

$$\forall k \in [1..n] \, \forall p \in [1..\ell] : \; P(\sigma, k, p) \in \{\lfloor f(p) \cdot k \rfloor - d, \lceil f(p) \cdot k \rceil + d\}.$$

We observe that PFAR is also NP-Hard as directly follows from the fact that unconstrained rank aggregation is NP-hard when $m \geq 4$ [1]. For some fixed $\alpha > 1$, We would like to find an algorithm that finds the p-fairest ranking among all rankings whose Kemeny distance from the set of input rankings is at most α times the minimum such distance.

Bi-criteria p-fair rank aggregation (BPFRA). The most general problem that we plan to consider in this context is the bi-criteria optimization problem, that is, for a given pair $(\alpha > 1, \beta > 1)$ and a set of m rankings find a ranking whose Kemeny distance to the set of rankings is at most α times the Kemeny distance of the aggregated rank from the set and its distance from a p-fair ranking is at most β, if such a ranking exists.

p-fair rank aggregation with affirmative action. We plan to consider a variant of p-fair rank aggregation that involves "affirmative action". This will be modeled by varying the proportion of the values of the protected attribute in the p-fair aggregated rank. For example, consider a binary protected attribute with values A and B each needs to appear the same number of times. Suppose that our goal is to promote the items with attribute value A. In this case we can vary the proportion of A making it higher in the top ranked elements and lower in the lower ranked elements so that overall items with attribute value A will appear the same number of times as items with attribute value B.

The complexity of individual p-fair ranking (IPF). We plan to further investigate the IPF problem for multi valued protected attributes as it is open whether it can be solved accurately in polynomial time. We conjecture that this problem is NP-Hard. We also plan to look for improved approximation algorithms for this problem.

Better approximation of rank aggregation subject to p-fairness (RAPF). We plan to develop more sophisticated RAPF algorithms with better approximation ratios, and to improve the computational aspects of the RAPF

problem. This problem can be formulated as an Integer Programming (IP) problem. We plan to consider various IP formulations as well as various rounding techniques to accelerate the computation.

Robust rank aggregation. It is known that rank aggregation under Kemeny distance is NP-hard. We will explore other aggregation methods, such as Spearman's footrule and Borda, and study how manipulable these rank aggregation methods are – that is, if only $x\%$ of the preferences are allowed to be changed, how easy it is to change the outcome.

4.2 Complex Constraints

Our goal is to optimize preference substitution to satisfy complex top-k fairness constraints, where the fairness requirement is defined over a set R of protected attributes. One of the objectives we will consider is minimizing the number of single ballot (ranking) substitutions that guarantee fairness in the top-k results. In a preliminary work we defined the problem of finding the smallest number of single ballot substitutions to promote a set of k candidates that satisfy fairness requirements defined over a set R of protected attributes to the top-k.

4.2.1 Research Directions

We assume that there are ℓ protected attributes , denoted $A_1, \ldots, A_\ell$. For $i \in [1..\ell]$, attribute A_i has ℓ_i possible values, denoted $A[i, j]$, for $j \in [1..\ell_i]$. Each candidate is associated with a specific value from each attribute. In addition, we are given target quantities $a[i, j]$, for $i \in [1..\ell]$, and $j \in [1..\ell_i]$, with property that all row marginals sum to k. Namely, for every $i \in [1..\ell]$, $\sum_{j=1}^{\ell_i} a[i, j] = k$. A fair outcome should satisfy the fairness condition that for $i \in [1..\ell]$, and $j \in [1..\ell_i]$, exactly $a[i, j]$ candidates whose A_i attribute value is $A[i, j]$ are elected.

We note that one way to approach this problem is by converting the multiple protected attributes to a single multi-valued protected attribute by computing joint distribution over the attributes assuming their independence. For example, instead of considering two binary valued attributes A_1 and A_2 we consider a single attribute with 4 possible values and the requirement that the value $i * j$ should appear $a[1, i] \cdot a[2, j]/k$ times, for $i, j \in \{1, 2\}$. The shortcomings of this approach are two-fold: First, this approach may yield that the problem is infeasible while there is still a solution without assuming independence. A solution that assumes independence may be inferior (require more substitutions) than a solution that does not assume independence.

In [20], we showed that the problem of finding the smallest number of single ballot substitutions (original preference) to promote a set of k candidates that satisfy proportionate representation over a single protected attribute is computationally easy for any domain size of the protected attribute. On the other hand the same problem becomes computationally hard if we increase the number of protected attributes. When there are two different protected attributes involved in outlining the fairness requirement, we proved that the decision version of that problem is (weakly) NP-hard, For three (or more) protected attribute, even the question whether there exists a set of top-k that satisfies the complex fairness constraint is strongly NP-Hard by a reduction from 3 Dimensional Matching. On the positive side for the case of two protected attributes we designed an efficient algorithm that obtains a 2 approximation factor and runs in $O(n^2 \ell \log m)$ time, where ℓ is the number of possible attribute values. We also designed an exact algorithm with running time n^c, where c is the size of the Cartesian product of all the attribute domains.

4.2.2 Open Problems

There propose two possible ways to extend these problems.

Improved approximation ratio in the case of 2 protected attributes. Since the problem of minimizing the number of single ballot substitutions in the case of 2 attributes is currently proven to be weakly NP-Hard, it may admit a PTAS (Polynomial Time Approximation Scheme). We plan to investigate the existence of a better approximation algorithm. Alternatively, we will try to improve the hardness result and show that this problem is strongly NP-Hard or Max-SNP Complete.

Relaxed solutions in the case of 3 or more protected attributes. Clearly, the hardness result of even checking the existence of a solution in case of 3 or more attributes precludes the existence of any approximation algorithm

for this case. We plan to design an algorithm that will generate a relaxed set of items/candidates. The relaxation may be in two dimensions: (i) the generated set will be a top-k set of candidates but the fairness requirements will not be fully satisfied for all protected attributes. (ii) the generated set will have size larger than k but it will satisfy the (lower bounds of the) fairness constraints for top k. Clearly, the larger the generated set is the easier the problem. We will find the smallest such extended set that guarantees the fairness constraints imposed by all protected attributes.

5 Multi Round Rank based Preference Aggregation

We study algorithmic challenges to satisfy output constraints in multi-round rank based preference aggregation methods, popularly known as ranked choice voting or (RCV) [12].

5.1 Research Directions

We start by describing the STV (single transferable vote) method [10, 13] that generalizes the IRV method, and selects a set of k items/candidates as the winners. STV is gaining popularity as an electoral system. It is used to elect candidates to the Australian Senate, in all elections in Malta, in most elections in the Republic of Ireland, and in Cambridge, MA. There are also plans to use STV in other USA localities. As mentioned in Section 3 this method of preference aggregation is also applicable in other settings.

The input to an STV preference aggregation method consists of m either complete or partial rankings of the items/candidates. Suppose that the total number of items/candidates is n out of which k items need to be elected. The preference aggregation process requires a predefined quota. In most cases this quota is Droop quota [22] defined as $\left\lfloor \frac{n}{k+1} \right\rfloor + 1$. The aggregation is done in rounds. In each round every item/candidate is associated a tally. Initially, the tally of every item is the number of rankings in which it is ranked highest. A round starts by considering the items whose tally is at least the quota. These items are elected in non-increasing order of their tally, as long as k items/candidates have not been elected (which always holds for Droop quota). When an item is elected their "surplus" (the number by which their tally exceeds the quota) is distributed to the next preferred item in their ranking (that has not been eliminated yet). The exact way this "surplus" is allocated varies. In a most cases, this allocation is done either fractionally or by a random selection of the surplus rankings out of all the rankings in which the elected item is top ranked. This is repeated as long as there are items whose tally is at least the quota (and k items/candidates have not been elected). Then, if less than k items/candidates are elected, the item/candidate with the smallest tally is eliminated from all the rankings, and the tallies are updated based on the new rankings. If the number of items/candidates remaining (not elected and not yet eliminated) equals the number of items/candidates left to be elected, these candidates are elected and the STV process terminates, otherwise the process repeats.

There is evidence that IRV and thus also STV preference aggregation methods are computationally hard to manipulate. It is NP-Hard to decide whether an IRV method can be manipulated even by adding one complete ranking [6]. On the positive side, [9, 11, 23] suggested branch and bound algorithms that use Integer Programming to compute the Margin of Victory (MOV) in IRV.

Approximating the number of ranking substitutions in multi round methods. We plan to develop approximation algorithms with proven performance for IRV and STV. The first step is to design such an algorithm for the simplest case which is approximating the minimum number of ranking substitutions required to change the outcome of an IRV preference aggregation method when every user is limited to input only two top items. From there we hope to be able to generalize to the IRV problem with no restriction on the ranking size, and eventually to the more general STV.

Improved computational frameworks for minimizing number of ranking substitutions in multi round methods. As mentioned above most of the existing computational frameworks are based on branch and bound algorithms. We plan to investigate other methods and possibly alternative formulation of the respective Integer Programming model that may result in more efficient computational frameworks.

Heuristic algorithms for minimizing the number of ranking substitutions in multi round methods. Another way to tackle the complex computational problem of minimizing number of ranking substitutions in multi round methods is designing heuristics for this task, analyzing and benchmarking these heuristics. One approach for designing such a heuristic for the problem of minimizing the number of ranking substitutions in STV to guarantee an elected set of k items with a given requirement on their protected attribute is by first identifying the desired elected set and then computing the number of substitutions required to achieve this set. One way of fixing the desired set is as follows. Run the STV process, and whenever the number of the currently elected items/candidates with a given value of their protected attribute reaches its bound, eliminate all the items/candidates with this value of their protected attribute. A naive implementation of this rule may not even guarantee a feasible solution and thus we also need to add the option of reintroducing items/candidates that were already eliminated. Analyzing such an algorithm is a challenge.

6 Non-rank based Preference Aggregation

Our goal is to study preference models that allow users to elicit their choice not as a ranked order. When the input preferences are not ranked, the output produces a set of k items that best reflects the users preferences. Akin to the previous two sections, our goal is to investigate which preference aggregation methods are suitable for such elicitation models, how to handle output constraints, and understand their computational implications. We identify the following research directions.

6.1 Research Directions

We begin by considering simple Boolean preference elicitation models, as"only likes", "likes and dislikes" , or "only dislikes". Indeed, such preference elicitation models are realistic in a wide variety of applications, such as providing preferences over products, news articles, movies, songs, social media posts, to name a few.

The simplest form of preference elicitation comes in the following form - each user $u(i)$ provides σ_i as preference, which is a Boolean vector of 1's and 0's over the set of n items, and the underlying application *only objective* is to find a set of k-items that are "most liked" by all the users. We propose to use Jaccard similarity or overlap similarity [24] for measuring similarity (inverse of distance) between two vectors in such cases. Given two vectors σ_i, σ_j their overlap similarity $S_{\sigma_i,\sigma_j} = \sum_{\forall \ell \in [n]} [\sigma_{i_\ell} \wedge \sigma_{j_\ell}]$, the number of positive bits that are shared between σ_i, σ_j. When the users provide both "likes" and "dislikes" and both have to be accounted for, we will use Hamming Distance which measures the minimum number of substitutions required to change σ_i to σ_j.

We have explored two alternative preference aggregation methods [3, 24] in the past that serve as the basis of this study.

- **Aggregated Voting.** Produce σ, such that $\mathcal{F}(\sigma, \sigma_1) + \mathcal{F}(\sigma, \sigma_2) + \ldots \mathcal{F}(\sigma, \sigma_m)$ is minimized.
- **Least Misery.** Produce σ, such that $\mathbf{Maximum}\{\mathcal{F}(\sigma, \sigma_1), \mathcal{F}(\sigma, \sigma_2), \ldots \mathcal{F}(\sigma, \sigma_m)\}$ is minimized.

The goal is to produce σ, which is also a vector of length n with exactly k number of 1's and remaining $0's$ that minimizes the Inverse of overlap similarity/Hamming Distance, denoted $\mathcal{F}(\cdot, \cdot)$, between σ and $\{\sigma_1, \sigma_2, \ldots\ldots, \sigma_m\}$.

We realize that the overlap similarity function is monotone, as when a new item is considered in the mix, the overlap similarity can never decrease (or inverse overlap similarity can never increase). This is likely to make preference aggregation computationally tractable and give rise to polynomial time solution to produce optimal σ. Under Hamming distance, however, finding σ considering either of the preference aggregation models is likely to be NP-hard, as a known NP-Complete problem Median String Problem could be reduced to a variant of this problem [14].

Satisfying Output Constraints. The output constraints in this case are defined on the top-k items/candidates and involve one or more protected attributes. When the output criteria is simple (designed on a single attribute), the Preference Aggregation Problems Considering Constraints defined in Section 3 for aggregated voting under Overlap Similarity is likely to give rise to computationally tractable problem for all three variants - Constrained

optimization, Optimization under budgetary constraints, and Bi-criteria optimization. On the other hand, these problems are likely to be computationally harder for least misery under Overlap Similarity. We will study how to exploit the monotonicity property of overlap similarity to see if it is possible to design greedy algorithms with provable approximation factors. Under Hamming Distance, irrespective of the underlying aggregation method, the Preference Aggregation Problems Considering Constraints are likely to be NP-hard, since the Preference Aggregation under Hamming Distance itself is NP-hard. We intend to study the possibility of designing approximation algorithms as well as efficient heuristics for these problems.

6.2 Open Problems

The applicability of the ordinal preference model is explored as one of the open problems - an ordinal value g is defined on an s-point performance scale, that is totally ordered $g_1 \prec g_2 \prec \ldots \prec g_s$. Given m input ordinal preferences and an output criteria, goal is to produce σ (an ordered list of n items/ top-k ordered/unordered set) that aggregates the preferences and satisfies the criteria. The input is studied as ordered sorting problem in decision aid literature [18]. Concretely speaking, each user's preference σ_i corresponds to assignment of each item into a pre-defined ordered categories, such as excellent, good, average, poor and the aggregation problem intends to find the best set of k-items σ. When studied under output constraints, the general challenge is to minimally change the original outcome so as to satisfy the constraints.

Preference Aggregation Methods. One key challenge in ordinal preference elicitation model is to identify the appropriate aggregation method and/or distance functions. Per our initial investigation, we realize that an ordinal preference elicitation could be expressed as a set of pairwise comparisons. As an example, if user $u(i)$ rates i_1 as excellent, i_2 as good, and i_3 as fair, this gives rise to the following 3 pairwise comparisons: $i_1 \prec i_2, i_2 \prec i_3, i_1 \prec i_3$. Given two preferences σ_i, σ_j, one can compute Kendall-Tau distance between these two to quantify the number of inversions or distance between them. Given m input preferences $\sigma_1, \sigma_2, \ldots \ldots, \sigma_m$, when the output is to produce an ordered outcome, the preference aggregation problem intends to produce a ranking σ that optimizes (minimizes) the Kemeny Distance [28] (sum of Kendall-Tau distance) between σ and $\{\sigma_1, \sigma_2, \ldots, \sigma_m\}$.

Additionally, we will study partial net score [18] of an item i ($PNS(i)$) that is proposed as an indicator of computing the overall "worth" of an item in decision aid literature. Based on the aforementioned pairwise representation, $PNS(i)$ can be expressed as $PNS(i) = \sum_{j \in [n] \setminus \{i\}} \left(|u^{[i \prec j]}| - |u^{[j \prec i]}| \right)$. Basically, $PNS(i)$ is the number of times item i is preferred over any other item j by any user (represented as $u^{[i \prec j]}$) minus the number of times these other items are preferred over i by any user (represented as $u^{[j \prec i]}$). By computing partial net score of each item one can design the outcome σ easily and efficiently. If σ needs to be ordered then the items will be ordered in decreasing order of partial net score; when the goal is to produce a top-k set of items, this will contain the items with the top-k highest partial net score.

Satisfying Output Constraints. We will study how to satisfy output constraints that are suitable to ordinal preference models. We will study both simple and complex output constraints, defined over single and multiple attributes, respectively. For the preference aggregation problem under output constraints, this is equivalent to producing a σ that minimizes the partial net score or Kemeny Distance between σ and input preferences, while satisfying the output constraints. When studied as an optimization problem under budgetary constraints B (B is the upper bound of partial net score or Kemeny Distance), the goal will be to produce σ, such that partial net score or Kemeny Distance is at most B and $\mathcal{C}$ is optimized. We anticipate most of these problems to be NP-hard. We will study how to design efficient approximation algorithms with provable guarantees, as well as effective heuristic algorithms.

7 Conclusion

The article lays a scientific foundation for systematically changing the outcome of a variety of preference aggregation methods to satisfy additional criteria related to fairness and robustness. The article studies single-round rank based, multi-round rank based, and non rank based preference aggregation methods that are suitable to different preference elicitation models and investigates how to minimally modify them to promote fairness. It identifies underlying computational and algorithmic challenges, proposes research directions, and formalizes several open problems.

8 Acknowledgment

The work is supported by the NSF CAREER Award #1942913, IIS #2007935, IIS #1814595, PPoSS:Planning #2118458, and by the Office of Naval Research Grants No: #N000141812838, #N000142112966.

References

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: ranking and clustering. Journal of the ACM (JACM), 55(5):1–27, 2008.

[2] S. Amer-Yahia, B. Omidvar-Tehrani, S. Basu, and N. Shabib. Group recommendation with temporal affinities. In International Conference on Extending Database Technology (EDBT), 2015.

[3] S. Amer-Yahia, S. B. Roy, A. Chawlat, G. Das, and C. Yu. Group recommendation: Semantics and efficiency. Proceedings of the VLDB Endowment, 2(1):754–765, 2009.

[4] M. Ayadi, N. B. Amor, J. Lang, and D. Peters. Single transferable vote: Incomplete knowledge and communication issues. In 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS 19), pages 1288–1296, 2019.

[5] P. C. Baayen and Z. Hedrlin. On the existence of well distributed sequences in compact spaces. Stichting Mathematisch Centrum. Zuivere Wiskunde, 1964.

[6] J. J. Bartholdi III and J. B. Orlin. Single transferable vote resists strategic voting. Social Choice and Welfare, 8(4):341–354, 1991.

[7] S. K. Baruah, N. K. Cohen, C. G. Plaxton, and D. A. Varvel. Proportionate progress: A notion of fairness in resource allocation. Algorithmica, 15(6):600–625, 1996.

[8] S. Basu Roy, L. V. Lakshmanan, and R. Liu. From group recommendations to group formation. In Proceedings of the 2015 ACM SIGMOD international conference on management of data, pages 1603–1616, 2015.

[9] M. Blom, P. Stuckey, and V. Teague. Toward computing the margin of victory in single transferable vote elections. INFORMS Journal on Computing, 31:636—653, 05 2019.

[10] M. Blom, P. J. Stuckey, and V. J. Teague. Toward computing the margin of victory in single transferable vote elections. INFORMS Journal on Computing, 31(4):636–653, 2019.

[11] M. Blom, V. Teague, P. J. Stuckey, and R. Tidhar. Efficient computation of exact irv margins. In Proceedings of the Twenty-Second European Conference on Artificial Intelligence, ECAI'16, pages 480—-488. IOS Press, 2016.

[12] D. Cary. Estimating the margin of victory for instant-runoff voting. EVT/WOTE, 11, 2011.

[13] A. Chakraborty, G. K. Patro, N. Ganguly, K. P. Gummadi, and P. Loiseau. Equality of voice: Towards fair representation in crowdsourced top-k recommendations. In Proceedings of the Conference on Fairness, Accountability, and Transparency, pages 129–138, 2019.

[14] C. de la Higuera and F. Casacuberta. Topology of strings: Median string is np-complete. Theoretical computer science, 230(1-2):39–48, 2000.

[15] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In Proceedings of the 3rd innovations in theoretical computer science conference, pages 214–226, 2012.

[16] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In Proceedings of the 10th international conference on World Wide Web, pages 613–622, 2001.

[17] A. M. Feldman and R. Serrano. Welfare economics and social choice theory. Springer Science & Business Media, 2006.

[18] J. Figueira, S. Greco, M. Ehrogott, P. Meyer, and M. Roubens. Choice, ranking and sorting in fuzzy multiple criteria decision aid. Multiple criteria decision analysis: State of the art surveys, pages 471–503, 2005.

[19] M. M. Islam, M. Asadi, and S. Basu Roy. Equitable top-k results for long tail data. Proceedings of the ACM on Management of Data, 1(4):1–24, 2023.

[20] M. M. Islam, D. Wei, B. Schieber, and S. B. Roy. Satisfying complex top-k fairness constraints by preference substitutions. Proceedings of the VLDB Endowment, 16(2):317–329, 2022.

[21] C. Kuhlman and E. Rundensteiner. Rank aggregation algorithms for fair consensus. Proceedings of the VLDB Endowment, 13(12):2706–2719, 2020.

[22] J. Lundell and I. Hill. Notes on the droop quota. Voting matters, 24:3–6, 2007.

[23] T. R. Magrino, R. L. Rivest, and E. Shen. Computing the margin of victory in IRV elections. In 2011 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 11), San Francisco, CA, 2011. USENIX Association.

[24] S. B. Roy, S. Thirumuruganathan, S. Amer-Yahia, G. Das, and C. Yu. Exploiting group recommendation functions for flexible preferences. In 2014 IEEE 30th international conference on data engineering, pages 412–423. IEEE, 2014.

[25] R. Tijdeman. The chairman assignment problem. Discrete Mathematics, 32(3):323–330, 1980.

[26] A. Van Zuylen and D. P. Williamson. Deterministic algorithms for rank aggregation and other ranking and clustering problems. In International Workshop on Approximation and Online Algorithms, pages 260–273. Springer, 2007.

[27] S. Verma and J. Rubin. Fairness definitions explained. In 2018 ieee/acm international workshop on software fairness (fairware), pages 1–7. IEEE, 2018.

[28] D. Wei, M. M. Islam, B. Schieber, and S. Basu Roy. Rank aggregation with proportionate fairness. In Proceedings of the 2022 International Conference on Management of Data, pages 262–275, 2022.

[29] M. Zehlike, K. Yang, and J. Stoyanovich. Fairness in ranking: A survey. arXiv preprint arXiv:2103.14000, 2021.

On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective

Jindong Wang[1,*], **Xixu Hu**[1,2‡,†], **Wenxin Hou**[3†], **Hao Chen**[4], **Runkai Zheng**[1,5‡], **Yidong Wang**[6], **Linyi Yang**[7], **Wei Ye**[6], **Haojun Huang**[3], **Xiubo Geng**[3], **Binxing Jiao**[3], **Yue Zhang**[7], **Xing Xie**[1]
[1]Microsoft Research, [2]City University of Hong Kong, [3]Microsoft STCA, [4]Carnegie Mellon University, [5]Chinese University of Hong Kong (Shenzhen), [6]Peking University, [7]Westlake University

Abstract

ChatGPT is receiving increasing attention over the past few months. While evaluations of various aspects of ChatGPT have been done, its robustness, i.e., the performance to unexpected inputs, is still unclear to the public. Robustness is of particular concern in responsible AI, especially for safety-critical applications. In this paper, we conduct a thorough evaluation of the robustness of ChatGPT from the adversarial and out-of-distribution (OOD) perspective. To do so, we employ the AdvGLUE and ANLI benchmarks to assess adversarial robustness and the Flipkart review and DDXPlus medical diagnosis datasets for OOD evaluation. We select several popular foundation models as baselines. Results show that ChatGPT shows consistent advantages on most adversarial and OOD classification and translation tasks. However, the absolute performance is far from perfection, which suggests that adversarial and OOD robustness remains a significant threat to foundation models. Moreover, ChatGPT shows astounding performance in understanding dialogue-related texts and we find that it tends to provide informal suggestions for medical tasks instead of definitive answers. Finally, we present in-depth discussions of possible research directions.

1 Introduction

Large language models (LLMs), or foundation models [7], have achieved significant performance on various natural language process (NLP) tasks. Given their superior in-context learning capability [30], prompting foundation models has emerged as a widely adopted paradigm of NLP research and applications. ChatGPT is a recent chatbot service released by OpenAI [33], which is a variant of the Generative Pre-trained Transformers (GPT) family. Thanks to its friendly interface and great performance, ChatGPT has attracted over 100 million users in two months.

Copyright 2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

[*]Contact: jindong.wang@microsoft.com.
[†]Equal contribution.
[‡]Work done during internship at Microsoft Research Asia.

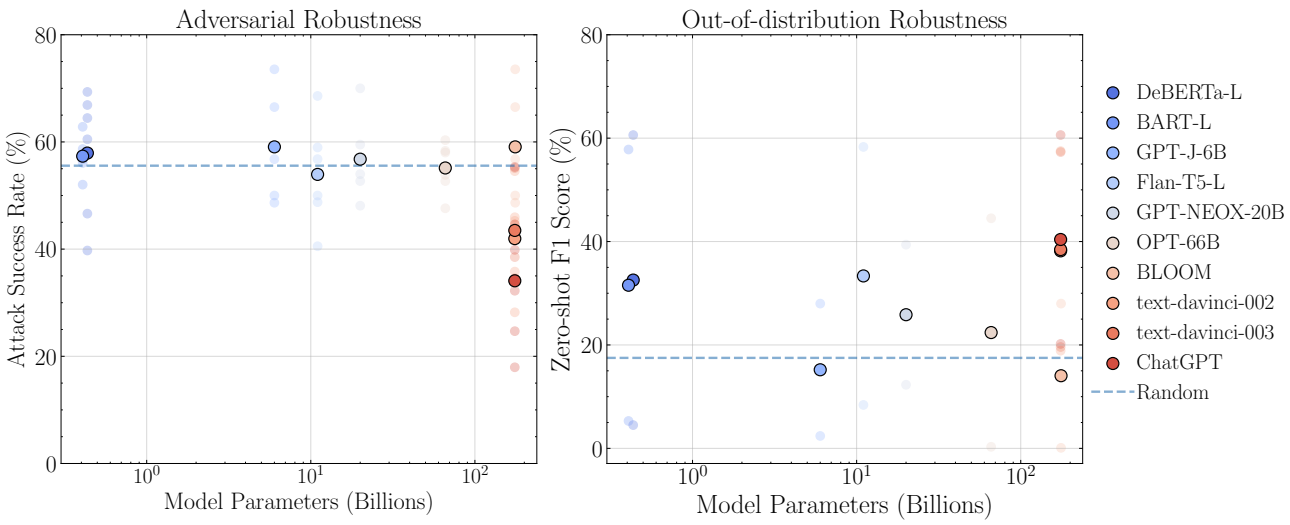


Figure 1: Robustness evaluation of different foundation models: performance vs. parameter size. Results show that ChatGPT shows consistent advantage on adversarial and OOD classification tasks. However, its absolute performance is far from perfection, indicating much room for improvement.

It is of imminent importance to evaluate the potential risks behind ChatGPT given its increasing worldwide popularity in diverse applications. While previous efforts have evaluated various aspects of ChatGPT in law [10], ethics [41], education [22], and reasoning [3], we focus on its robustness [4], which, to our best knowledge, has not been thoroughly evaluated yet. Robustness refers to the ability to withstand disturbances or external factors that may cause it to malfunction or provide inaccurate results. It is important to practical applications especially the safety-critical scenarios. For instance, if we apply ChatGPT or other foundation models to fake news detection, a malicious user might add noise or certain perturbations to the content to bypass the detection system. Without robustness, the reliability of the system collapses.

Robustness threats exist in a wide range of scenarios: out-of-distribution (OOD) samples [55], adversarial inputs [15], long-tailed samples [60], noisy inputs [31], and many others. In this paper, we pay special attention to two popular types of robustness: the adversarial and OOD robustness, both of which are caused through input perturbation. Specifically, adversarial robustness studies the model's stability to adversarial and imperceptible perturbations, e.g., adding trained noise to an image or changing some keywords of a text. On the other hand, OOD robustness measures the performance of a model on unseen data from different distributions of the training data, e.g., classifying sketches using a model trained for art painting or analyzing a hotel review using a model trained for appliance review. More background of these robustness is elaborated in section 2.2.

Zero-shot robustness evaluation. While robustness research often requires training and optimization (e.g., fine-tuning, linear probing, domain adaptation and generalization, section 2.2), in this work, we focus on zero-shot robustness evaluation. Given a foundation model, we perform inference directly on the test dataset for evaluation. We argue that it becomes more expensive and unaffordable to train, or even load existing (and future, larger) foundation models. For instance, the largest Flan-T5 model has 11 billion parameters [12], which is already beyond the capability of most researchers and practitioners. Thus, their zero-shot performance becomes important to downstream tasks. On the other hand, foundation models are typically trained on huge volumes of datasets with huge amount of parameters, which seems to challenge conventional machine learning theories:

Are large foundation models all we need for robustness?

In this work, we conduct a thorough evaluation of ChatGPT on its adversarial and OOD robustness for natural language understanding tasks. It is challenging to select appropriate datasets for evaluating ChatGPT since it

is known to be trained on huge text datasets as of 2021. Eventually, we leverage several recent datasets for our evaluation: AdvGLUE [54] and ANLI [32] for adversarial robustness and two new datasets for OOD robustness: Flipkart review [49] and DDXPlus medical diagnosis datasets [46]. Furthermore, we randomly selected 30 samples from AdvGLUE to form an adversarial translation dataset to evaluate the translation performance. These datasets represent various levels of robustness, thus provide a fair evaluation. The detailed information of these datasets are introduced in section 3. We then select several popular foundation models from Huggingface model hub and OpenAI service[1] to compare with ChatGPT. In summary, we have 9 tasks and 2,089 test examples.

Our findings. We perform zero-shot inference on all tasks using these models and fig. 1 summarizes our main results. The major findings of the study include:

1. What ChatGPT does well:
    - ChatGPT shows consistent improvements on most adversarial and OOD classification tasks.
    - ChatGPT is good at translation tasks. Even in the presence of adversarial inputs, it can consistently generate readable and reasonable responses.
    - ChatGPT is better at understanding dialogue-related texts than other foundation models. This could be attributed to its enhanced ability as a chatbot service, leading to good performance on DDXPlus dataset.
2. What ChatGPT does not do well:
    - The absolute performance of ChatGPT on adversarial and OOD classification tasks is still far from perfection even if it outperforms most of the counterparts.
    - The translation performance of ChatGPT is worse than its instruction-tuned sibling model text-davinci-003.
    - ChatGPT does not provide definitive answers for medical-related questions, but instead offers informed suggestions and analysis. Thus, it can serve as a friendly assistant.
3. Other general findings about foundation models:
    - Task-specific fine-tuning helps language models perform better on related tasks, e.g., NLI-fine-tuned RoBERTa-L has similar performance to Flan-T5-L.
    - Instruction tuning benefits large language models, e.g., Flan-T5-L achieves comparable performance to text-davinci-002 and text-davinci-002 with significantly less parameters.

Beyond evaluations, we share more reflections in the discussion and limitation sections, providing experience and suggestions to future research. Finally, we open-source our code and results at `https://github.com/microsoft/robustlearn` to facilitate future explorations.

2 Background

2.1 Foundation Models, ChatGPT, and Existing Evaluation

Foundation models have become a popular research and application paradigm for natural language process tasks. Since foundation models are trained on large volumes of data, they show significant performance improvement on different downstream tasks such as sentiment analysis, question answering, automatic diagnosis, logical reasoning, and sequence tagging. ChatGPT is a generative foundation model that belongs to the GPT-3.5 series

[1] Huggingface: `https://huggingface.co/models`. OpenAI service: `https://openai.com/api`.

in OpenAI's GPT family, coming after GPT [37], GPT-2 [38], GPT-3 [8], and InstructGPT [34]. In contrast to its predecessors, ChatGPT makes it easy for every one to use just through a browser with enhanced multi-turn dialogue capabilities. Although the technical details of ChatGPT is still not released, it is known to be trained using reinforcement learning from human feedback (RLHF) [11] with instruction tuning. Other than natural language processing, there are also emerging efforts in building foundation models for computer vision [13], music generation [1], biology [23, 25], and speech recognition [36].

Previous efforts evaluate ChatGPT in different aspects [50]. [3] proposes a multi-task, multi-modal, and multilingual evaluation of ChatGPT on different tasks. They showed that ChatGPT performs reasonably well on most tasks, while it does not bring great performance on low-resource tasks. Similar empirical evaluations are also made by [2, 16]. Specifically, [35] also did several evaluations and they found that ChatGPT does not do well on fine-grained downstream tasks such as sequence tagging. In addition to research from artificial intelligence, researchers from other areas also showed interest in ChatGPT. [18, 41] expressed concerns that ChatGPT and other large models should be regulated since they are double-edged swords. The evaluations on ethics are done in [62]. There are reflections and discussions from law [10], education [17, 22, 26, 44], human-computer interaction [45], medicine [21], and writing [5]. To the best of our knowledge, a thorough robustness evaluation is currently under-explored.

2.2 Robustness

In the following, we present the formulation of robustness with the classification task (other tasks can be formulated similarly). We are given a K-class classification dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, where $\mathbf{x} \in \mathbb{R}^d$ and $y \in [K]$ are its d-dimensional input and output, respectively. We use $\ell[\cdot, \cdot]$ to denote the loss function.

Adversarial robustness An adversarial input [15] $\mathbf{x}'$ is generated by adding a ϵ-bounded, imperceptible perturbation δ to the original input $\mathbf{x}$. The optimal classifier can be learned by optimizing the following objective [27]:

$$\min_{f \in \mathcal{H}} \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}} \max_{|\delta|\leq\epsilon} \ell[f(\mathbf{x}+\delta), y].$$

Out-of-distribution robustness On the other hand, OOD robustness (generalization) [42, 55] aims to learn an optimal classifier on an unseen distribution by training on existing data. One popular formulation for OOD robustness is to minimize the average risk on all distributions e, which is sampled over the set of all possible distributions (could be large than $\mathcal{D}$):

$$\min_{f \in \mathcal{H}} \mathbb{E}_{e\sim\mathcal{Q}} \mathbb{E}_{(\mathbf{x},y)\in\mathcal{D}^e} \ell[f(\mathbf{x}), y].$$

[57] presented GLUE-X, a benchmark based on GLUE and then conducted a thorough evaluation of the OOD robustness of language models by training on in-distribution (ID) sets and then testing on OOD sets. Ours, however, performs zero-shot evaluation. The OOD robustness of ChatGPT cannot be evaluated by GLUE and GLUE-X benchmarks since it may include the entire GLUE datasets in its training data.

3 Datasets and Tasks

3.1 Adversarial Datasets

We adopt AdvGLUE [54] and adversarial natural language inference (ANLI) [32] benchmarks for evaluating adversarial robustness. AdvGLUE is a modified version of the existing GLUE benchmark [52] by adding different kinds of adversarial noise to the text: word-level perturbation (typo), sentence-level perturbation (distraction),

Table 2: Statistics of test sets in this paper

Area	Dataset	Task	#Sample	#Class
Adversarial robustness	SST-2	sentiment classification	148	2
	QQP	quora question pairs	78	3
	MNLI	multi-genre natural language inference	121	3
	QNLI	question-answering NLI	148	2
	RTE	textual entailment recognition	81	2
	ANLI	text classification	1200	3
	AdvGLUE-T	machine translation (En $\rightarrow$ Zh)	30	-
OOD robustness	Flipkart	sentiment classification	331	2
	DDXPlus	medical diagnosis classification	100	50

and human-crafted perturbations. We adopt 5 tasks from AdvGLUE: SST-2, QQP, MNLI, QNLI, and RTE. Since the test set of AdvGLUE is not public, we adopt its development set instead for evaluation. Although AdvGLUE is a classification benchmark, we additionally construct an adversarial machine translation (En $\rightarrow$ Zh) dataset, termed AdvGLUE-T, by randomly selecting 30 samples from AdvGLUE.

ANLI is a large-scale dataset designed to assess the generalization and robustness of natural language inference (NLI) models, which was created by Facebook AI Research. It comprises 16,000 premise-hypothesis pairs that are classified into three categories: entailment, contradiction, and neutral. The dataset is divided into three parts (R1, R2, and R3) based on the number of iterations used during its creation, with R3 being the most difficult and diverse. Therefore, we select the test set of R3 for evaluating the adversarial robustness of our models.

3.2 Out-of-distribution Datasets

We adopt two new datasets[2] for OOD robustness evaluation: Flipkart [49] and DDXPlus [46]. Flipkart is a product review dataset and DDXPlus is a new medical diagnosis dataset, both of which are released in 2022. These two datasets can be used to construct classification tasks. We randomly sample a subset of each dataset to form the test sets. table 2 shows the statistics of each dataset.

Remark: Finding an OOD dataset for large models like ChatGPT is difficult due to the unavailability of its training data. Consider these datasets as 'out-of-example' datasets since they did not show up in ChatGPT's training data. Additionally, distribution shift may happen at different dimensions: not only across domains, but also across time. Thus, even if ChatGPT and other LLMs may already use similar datasets like medical diagnosis and product review, our selected datasets are still useful for OOD evaluation due to temporal distribution shift. Finally, we must admit the limitation of these datasets and look forward to brand new ones for more thorough evaluation.

[2]Considering ChatGPT is reported to be trained on a substantial corpus of internet language data as of 2021, identifying an out-of-distribution dataset poses a difficulty. Furthermore, we concern that previous natural language processing datasets predating 2022 may have been assimilated by ChatGPT, so we only utilize datasets that are recently released.

Table 3: Zero-shot classification results on adversarial (ASR↓) and OOD (F1↑) datasets. The best and second-best results are highlighted in **bold** and underline.

Model & #Param.	Adversarial robustness (ASR↓)						OOD robustness (F1↑)	
	SST-2	QQP	MNLI	QNLI	RTE	ANLI	Flipkart	DDXPlus
Random	50.0	50.0	66.7	50.0	50.0	66.7	20.0	4.0
DeBERTa-L (435 M)	66.9	39.7	64.5	46.6	60.5	69.3	**60.6**	4.5
BART-L (407 M)	56.1	62.8	58.7	52.0	56.8	57.7	57.8	5.3
GPT-J-6B (6 B)	48.7	59.0	73.6	50.0	56.8	66.5	28.0	2.4
Flan-T5-L (11 B)	40.5	59.0	48.8	50.0	56.8	68.6	58.3	8.4
GPT-NEOX-20B (20 B)	52.7	56.4	59.5	54.0	48.1	70.0	39.4	12.3
OPT-66B (66 B)	47.6	53.9	60.3	52.7	58.0	58.3	44.5	0.3
BLOOM (176 B)	48.7	59.0	73.6	50.0	56.8	66.5	28.0	0.1
text-davinci-002 (175 B)	46.0	28.2	54.6	45.3	35.8	68.8	57.5	18.9
text-davinci-003 (175 B)	44.6	55.1	44.6	38.5	34.6	62.9	57.3	19.6
ChatGPT (175 B)	**39.9**	**18.0**	**32.2**	**34.5**	**24.7**	**55.3**	**60.6**	**20.2**

4 Experiment

4.1 Zero-shot Classification

4.1.1 Setup

We compare the performance of ChatGPT on AdvGLUE classification benchmark with the following existing popular foundation models: DeBERTa-L [19], BART-L [24], GPT-J-6B [53], Flan-T5 [12, 39], GPT-NEOX-20B [6], OPT-66B [59], BLOOM [40], and GPT-3 (text-davinci-002 and text-davinci-003) [3]. The latter two are from OpenAI API service and the rest are on Hugging face model hub. The notation '-L' means '-large', as we only evaluate the large version of these models.

For adversarial classification tasks on AdvGLUE and ANLI, we adopt attack success rate (ASR) as the metric for robustness. For OOD classification tasks, F1-score (F1) is adopted as the metric. As mentioned before, we only perform zero-shot evaluation. Thus, we simply run all models on a local computer with plain GPUs, which could be the case in most downstream applications.[4] Note that we use the NLI-fine-tuned version of DeBERTa-L and BART-L on natural language inference tasks to perform zero-shot classification since they are not originally designed for text classification. For other models, we adopt the prompt-based paradigm to get answers for classification by inputting prompts. Note that we manually processed some outputs since the outputs of some generative LLMs are not easy to control.

4.1.2 Results

The classification results of adversarial and OOD robustness are shown in table 3.

First, **ChatGPT shows consistent improvements on adversarial datasets.** It outperforms all counterparts on all adversarial classification tasks. However, we see that there is still room for improvement since the absolute performance is far from perfection. For instance, the ASRs on SST-2 and ANLI are 40% and 55.3%, respectively, indicating much room for improvement. This could be due to the reason that they are trained on clean corpus and

[3] Note that the classification task may be unfavorable to the generative models since we did not limit their output space as discriminative models like DeBERTa-L do.

[4] Even the local computer is not that "plain" since it requires at least 1 A100 GPU with 80 GB of memory.

some adversarial texts are washed out from the training data. Beyond ChatGPT, it is also surprising to find that most methods only achieve slightly better than random guessing, while some even do not beat random guessing. This indicates that the zero-shot adversarial robustness of most foundation models is not promising. In addition to foundation models, we are surprised to find that some small models also achieve great performance on adversarial tasks while it has much less parameters than the strong models (e.g, DeBERTa-L on QQP and QNLI tasks). This indicate that fine-tuning on relevant tasks can still improve the performance. Furthermore, Flan-T5 also achieves comparable performance to most larger models. Since Flan-T5 is also trained via instruction tuning, this implies the efficacy of such training approach in prompting-based NLP tasks.

Second, **all models after GPT-2 (text-davinci-002, text-davinci-003, and ChatGPT) perform well on OOD datasets.** This observation is in consistency with recent finding in OOD research that the in-distribution (ID) and OOD performances are positively correlated [29]. However, ChatGPT and its sibling models perform much better on DDXPlus, indicating its ability to recognize new or diverse domain data. Additionally, some large models performs better, e.g., Flan-T5-L outperforms some larger models such as OPT-66B and BLOOM. This can be explained as overfitting on certain large models or they have an inverse ID-OOD relation [47] on our test sets. It should also be noted that the absolute performance of ChatGPT and davinci series are still far from perfection.

Third, on the DDXPlus dataset, **ChatGPT is better at understanding diaglogue-related texts compared with other LLMs.** The DDXPlus benchmark presents a formidable challenge for many models. The majority of models perform at a level akin to random chance, with the exception of the davinci series and ChatGPT, which exhibit exceptional performance. One plausible explanation for the superior performance of these three models may be their substantial increase in the number of model parameters. This substantial increase in parameter count may enable the model to learn more complex representations and subsequently result in an improvement of performance. Another possible reason for the success of ChatGPT is its ability to understand the conversational context of DDXPlus, which consists of doctor-posed diagnostic questions and patient responses. ChatGPT has demonstrated superior performance in understanding conversational context compared to previous models, which likely contributes to its improved performance on this dataset.

Finally, it is worth noting that due to the critical nature of the healthcare field, **ChatGPT does not provide definitive answers in medical-related questions but instead offers informed suggestions and analysis, followed by a recommendation for further offline testing and consultation to ensure accurate diagnosis.** When the provided information is insufficient to make a judgment, ChatGPT will acknowledge this and offer an explanation, demonstrating its responsible approach to medical-related inquiries. This highlights the benefits of using ChatGPT for medical-related inquiries compared to search engines, as it can provide comprehensive analysis and suggestions without requiring the users to have medical expertise, while also being responsible and cautious in its responses. This suggests a promising future for the integration of ChatGPT in computer-aided diagnosis systems.

4.2 Zero-shot Machine Translation

4.2.1 Setup

We further evaluate the adversarial robustness of ChatGPT on an English-to-Chinese (En $\rightarrow$ Zh) machine translation task. The test set (AdvGLUE-T) is sub-sampled from the adversarial English text in AdvGLUE and we manually translate them into Chinese as ground truth. We evaluate the zero-shot translation performance of ChatGPT against text-davinci-002 and text-davinci-003. We further adopt two fine-tuned machine translation models from the Huggingface model hub: OPUS-MT-EN-ZH [48] and Trans-OPUS-MT-EN-ZH[5]. We report

[5]Note that there are only few En $\rightarrow$ Zh machine translation models released on Huggingface model hub and we pick the top two with the most downloads.

BLEU, GLEU, and METEOR in experiments to conduct a fair comparison among several models.[6]

4.2.2 Results

The results of zero-shot machine translation are shown in table 4. Note that all three models from the GPT family outperforms the fine-tuned models. Interestingly, text-davinci-003 generalizes the best on all metrics. The performance of ChatGPT is better to text-davinci-002 on BLUE and GLUE, but slightly worse on METOR. While differing in metrics, we find **the translated texts of ChatGPT (and text-davinci-002 and text-davinci-003) is very readable and reasonable to humans, even given adversarial inputs.** This indicates the adversarial robustness capability on machine translation of ChatGPT might originate from GPT-3.

Table 4: Zero-shot machine translation results on adversarial text sampled from AdvGLUE.

Model	BLEU↑	GLEU↑	METOR↑
OPUS-MT-EN-ZH	18.11	26.78	46.38
Trans-OPUS-MT-EN-ZH	15.23	24.89	45.02
text-davinci-002	24.97	36.30	<u>59.28</u>
text-davinci-003	**30.60**	**40.01**	**61.88**
ChatGPT	<u>26.27</u>	<u>37.29</u>	58.95

4.3 Case Study

table 5 shows some results of ChatGPT across word-level (typo) and sentence-level (distraction) adversarial inputs. It is evident that both adversaries pose a considerable challenge to ChatGPT, through their ability to mislead the model's judgement. It should be noted that these adversaries are prevalent in everyday interactions, and the existence of numerous forms of textual adversarial attacks highlights the necessity of defensive strategies for ChatGPT. Unlike adversarial inputs, it is not easy to analyze why ChatGPT performs bad for OOD datasets since the notion of "distribution" is hard to quantify.

5 Discussion

5.1 Adversarial Attack Remains a Major Threat

As discussed in experiments, dealing with adversarial inputs still remains challenging to large foundation models. With the proliferation of foundation model service such as ChatGPT, such adversarial vulnerability remains a major threat to various downstream scenarios, especially those safety-critical applications. On the other hand, since adversarial inputs are subjectively generated by humans, but not exist in nature, we argue that foundation models might never cover all distributions of possible adversarial inputs during their training [20]. Other than error correction, a possible solution for model owners is to first inject adversarial inputs to their training data, which could improve its robustness to existing adversarial noise. Then, as a long-standing goal to improve the model robustness, the pre-trained model can be continuously trained on human-generated or algorithm-generated adversarial inputs.

As for those who cannot train large models and only use them in downstream tasks, such threat still exists due to the defect inheritance of pre-trained models. In this case, how to achieve perfect fine-tuning or adaptation performance on downstream tasks while certainly reducing the defect inheritance remains a major challenge. Luckily, some pioneering work [9, 61] might provide solutions. This represents a novel and emerging direction for

[6]We use NLTK (`https://www.nltk.org/`) to calculate these metrics.

Table 5: Case study on adversarial examples. Adversarial manipulations are marked red.

Type	Input	Truth	davinci003	ChatGPT
word-level (typo)	i think you 're here for raunchy college humor .	Positive	Negative	Negative
	Mr. Tsai is a very oriignal artist in his medium , and what time is it there?	Positive	Positive	Positive
	Q1: Can you TRANSLATE these to English language? Q2: Cn you translate ths from Bengali to English lagnuage?	Not equivalent	Not equivalent	Equivalent
	Q1: What are the best things in Hog Kong? Q2: What is the best thing in Hong Kong?	Equivalent	Not equivalent	Not equivalent
sentence-level (distraction)	Question: What is the minimum required if you want to teach in Canada? Sentence: @KMcYo0 In most provinces a second Bachelor's Degree such as a Bachelor of Education is required to become a qualified teacher.	Not entailment	Entailment	Entailment
	Question: @uN66rN What kind of water body is rumored to be obscuring Genghis Khan's burial site? Sentence: Folklore says that a river was diverted over his grave to make it impossible to find (the same manner of burial as the Sumerian King Gilgamesh of Uruk and Atilla the Hun).	Entailment	Not entailment	Not entailment
	https://t.co/1GPp0U the iditarod lasts for days - this just felt like it did .	Negative	Positive	Negative
	holden caulfield did it better . https://t.co/g4vJKP	Negative	Positive	Negative

future research. However, as foundation models grow larger that go beyond the capabilities of most researchers, reducing the defects through fine-tuning could be impossible. An open question rises for both model owners and downstream users on how to defend the adversarial attack.

In addition to adversaries in training data, prompts can also be attacked [28], which requires further knowledge and algorithms to deal with. This is currently a challenging problem due to the sensitivity of prompting to LLMs.

5.2 Can OOD Generalization be Solved by Large Foundation Models?

Larger models like ChatGPT and text-davinci-003 have the potential to achieve superior performance on OOD datasets with better prompt engineering, inspiring us to think of the problem: is OOD generalization solved by these giant models? The huge training data and parameters are a double-edged sword: overfitting vs. generalization. It is also intuitive to think that OOD data is unseen during training, so adding it into training set is enough, which is what these large models did. Is the "unreasonable effectiveness of data" [43] real? However, as the model sizes are becoming larger, it still remains unknown when and why LLMs will overfit.

Another possible reason is the training data of ChatGPT and text-davinci-003 actually encompass similar distributions to our test sets even if they are collected after 2021. Flipkart is for product review and DDXPlus is for medical diagnosis, which in fact are common domains widely existing on the Internet. Thus, they could be not OOD to these models, that could lead to overfitting. New datasets from long-tailed domains are in need for more fair evaluations.

Finally, our analysis does not show that ID-OOD performances are always positively correlated [29], but can sometimes inversely correlated [47]. Regularization and other techniques should be developed to improve the OOD performance of language models.

5.3 Beyond NLP Foundation Models

Adversarial and OOD robustness do not only exist in natural language, but also in other domains. In fact, most research comes from machine learning and computer vision communities. Researchers in computer vision area could possibly think: can we solve OOD and adversarial robustness in image data by training a vision foundation model? For instance, the recent ViT-22B [13] scales vision Transformer [14] to 22 billion parameters by training it on the 4 billion JFT dataset [58] (a larger version of the previous JFT-300M dataset [43]), which becomes the largest vision foundation model to date. ViT-22B shows superior performance on different image classification tasks. However, it does not show "emergent abilities" [56] with the increment of parameters as other LLMs. Not only LLMs, the robustness in other areas also remains to be solved.

Back to theory, algorithms, and optimization areas, which foundational research areas in artificial intelligence. Will the large foundation models disrupt these areas? First, we should acknowledge that the success of foundation models should also attribute to these areas, e.g., most LLMs adopt the Transformer [51] and other advanced learning and training research. Second, the success of foundation models shed light on these areas: can we solve the problems like adversarial and OOD by developing new theories, algorithms, and optimization methods? Such research could offer valuable contribution to foundation models, e.g., improve the data and training efficiency and efficacy. Finally, researchers in these areas should not be dis-encouraged since the advance of scientific research should be diverse and not restricted to those done with rich computing resources.

6 Limitation

This paper offers a preliminary empirical study on the robustness of large foundation models, which has the following limitations.

First, we only perform zero-shot classification using ChatGPT and other models. Results of these models could change if we perform fine-tuning or adaptation given enough computing resources. But as we stated in introduction, it is expensive and un-affordable to perform further operation on today's latest foundation models, we believe zero-shot evaluation is reasonable.

Second, it seems controversial to evaluate large foundation models on small datasets in this work. However, since the training data of ChatGPT and some large models remains unclear, it is difficult to find larger datasets. Especially, ChatGPT is trained on huge datasets on the Internet as of 2021, making it more difficult to find appropriate datasets for thorough evaluation. We do believe more datasets can be used for such evaluation.

Third, we did most evaluations on text classification and only minor evaluations on machine translation. It is well-known that ChatGPT and other foundation models can do more tasks such as generation. Again, because of lack of appropriate datasets, evaluating generation performance is also difficult. We also admit that introducing more proper prompts could improve its performance.

Fourth, it is worth noting that ChatGPT is mainly designed to be a chatbot service rather than a tool for text classification. Our evaluations are mainly for classification, which have nothing to do with the robustness of ChatGPT for online chatting experience. We do hope every end-user can find ChatGPT helpful in their lives.

Finally, we could further provide detailed synopsis by conducting experiments on data before 2021 as comparisons and analyzing more OOD cases to see why ChatGPT succeeds or fails. Other experiments include detailed ablation study using different language models and investigation of induced outputs by LLMs through prompts. These can be done in future work. Another claim is that ChatGPT is not perfect for adversarial tasks. But we also need to develop certain metrics to show 'how good' is the performance.

7 Conclusion

This paper presented a preliminary evaluation of the robustness of ChatGPT from the adversarial and out-of-distribution perspective. While we acknowledge the advance of large foundation models on adversarial and out-of-distribution robustness, our experiments show that there is still room for improvement to ChatGPT and other large models on these tasks. Afterwards, we presented in-depth analysis and discussion beyond NLP area, and then highlight some potential research directions regarding foundation models. We hope our evaluation, analysis, and discussions could provide experience to future research.

Acknowledgement

This paper received attentions from many experts since its first version was released on ArXiv. Authors would like to thank all who gave constructive feedback to this work.

Disclaimer

Potential Ethics and Societal Concerns raised by ChatGPT Robustness The increasing popularity of ChatGPT and other chatbot services certainly face some new concerns from both ethics and society. The purpose of this paper is to show that ChatGPT can be attacked by adversarial and OOD examples using existing public dataset, but not to attack it intentionally. We hope that this will not be leverage by end-users. Finally, we also hope the community can realize the importance of robustness research and develop new technologies to make our systems more secure, robust, and responsible.

The contribution of each author Jindong led the project, designed experiments, wrote the code framework, and wrote the paper. Xixu and Wenxin shared equal contributions. Xixu was in charge of processing, experimenting, and writing about the DDXPlus and ANLI datasets. Wenxin designed all prompts to generative models and wrote about this part. Hao did the machine translation experiments, wrote necessary codes, and was in charge of code organization and reproducibility. Runkai helped polish the paper and organized case study. Other authors actively participated in this project from day one, reviewed the paper carefully, and provided valuable comments to improve this work.

References

[1] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. arXiv preprint arXiv:2301.11325, 2023.

[2] Amos Azaria. Chatgpt usage and limitations. 2022.

[3] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. arXiv preprint arXiv:2302.04023, 2023.

[4] Yoshua Bengio, Yann Lecun, and Geoffrey Hinton. Deep learning for ai. Communications of the ACM, 64(7):58–65, 2021.

[5] Som Biswas. Chatgpt and the future of medical writing, 2023.

[6] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022.

[7] 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. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258, 2021.

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.

[9] Ting-Wu Chin, Cha Zhang, and Diana Marculescu. Renofeation: A simple transfer learning method for improved adversarial robustness. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3243–3252, 2021.

[10] Jonathan H Choi, Kristin E Hickman, Amy Monahan, and Daniel Schwarcz. Chatgpt goes to law school. Available at SSRN, 2023.

[11] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. Advances in neural information processing systems, 30, 2017.

[12] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. arXiv preprint arXiv:2210.11416, 2022.

[13] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. arXiv preprint arXiv:2302.05442, 2023.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[15] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

[16] Roberto Gozalo-Brizuela and Eduardo C Garrido-Merchan. Chatgpt is not all you need. a state of the art review of large generative ai models. arXiv preprint arXiv:2301.04655, 2023.

[17] Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. arXiv preprint arXiv:2301.07597, 2023.

[18] Philipp Hacker, Andreas Engel, and Marco Mauer. Regulating chatgpt and other large generative ai models. arXiv preprint arXiv:2302.02337, 2023.

[19] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. arXiv preprint arXiv:2006.03654, 2020.

[20] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. Advances in neural information processing systems, 32, 2019.

[21] Katharina Jeblick, Balthasar Schachtner, Jakob Dexl, Andreas Mittermeier, Anna Theresa Stüber, Johanna Topalis, Tobias Weber, Philipp Wesp, Bastian Sabel, Jens Ricke, et al. Chatgpt makes medicine easy to swallow: An exploratory case study on simplified radiology reports. arXiv preprint arXiv:2212.14882, 2022.

[22] Mohammad Khalil and Erkan Er. Will chatgpt get you caught? rethinking of plagiarism detection. arXiv preprint arXiv:2302.04335, 2023.

[23] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36(4):1234–1240, 2020.

[24] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880, 2020.

[25] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. Biogpt: generative pre-trained transformer for biomedical text generation and mining. Briefings in Bioinformatics, 23(6), 2022.

[26] Muneer M Alshater. Exploring the role of artificial intelligence in enhancing academic performance: A case study of chatgpt. Available at SSRN, 2022.

[27] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083, 2017.

[28] Natalie Maus, Patrick Chao, Eric Wong, and Jacob Gardner. Adversarial prompting for black box foundation models. arXiv preprint arXiv:2302.04237, 2023.

[29] John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In International Conference on Machine Learning, pages 7721–7735. PMLR, 2021.

[30] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? arXiv preprint arXiv:2202.12837, 2022.

[31] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. Advances in neural information processing systems, 26, 2013.

[32] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020.

[33] OpenAI. `https://chat.openai.com.chat`, 2023.

[34] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. arXiv preprint arXiv:2203.02155, 2022.

[35] Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. Is chatgpt a general-purpose natural language processing task solver? arXiv preprint arXiv:2302.06476, 2023.

[36] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. arXiv preprint arXiv:2212.04356, 2022.

[37] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. In Advances in neural information processing systems, pages 8735–8745, 2018.

[38] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9, 2019.

[39] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 21(1):5485–5551, 2020.

[40] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. arXiv preprint arXiv:2211.05100, 2022.

[41] Yiqiu Shen, Laura Heacock, Jonathan Elias, Keith D Hentel, Beatriu Reig, George Shih, and Linda Moy. Chatgpt and other large language models are double-edged swords, 2023.

[42] Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. arXiv preprint arXiv:2108.13624, 2021.

[43] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In Proceedings of the IEEE international conference on computer vision, pages 843–852, 2017.

[44] Teo Susnjak. Chatgpt: The end of online exam integrity? arXiv preprint arXiv:2212.09292, 2022.

[45] Wilbert Tabone and Joost de Winter. Using chatgpt for human–computer interaction research: A primer. 2023.

[46] Arsene Fansi Tchango, Rishab Goel, Zhi Wen, Julien Martel, and Joumana Ghosn. Ddxplus: A new dataset for automatic medical diagnosis. Proceedings of the Neural Information Processing Systems-Track on Datasets and Benchmarks, 2, 2022.

[47] Damien Teney, Yong Lin, Seong Joon Oh, and Ehsan Abbasnejad. Id and ood performance are sometimes inversely correlated on real-world datasets. arXiv preprint arXiv:2209.00613, 2022.

[48] Jörg Tiedemann and Santhosh Thottingal. OPUS-MT — Building open translation services for the World. In Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT), Lisbon, Portugal, 2020.

[49] Nirali Vaghani and Mansi Thummar. Flipkart product reviews with sentiment dataset, 2023.

[50] Eva AM van Dis, Johan Bollen, Willem Zuidema, Robert van Rooij, and Claudi L Bockting. Chatgpt: five priorities for research. Nature, 614(7947):224–226, 2023.

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.

[52] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019. In the Proceedings of ICLR.

[53] Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

[54] Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. arXiv preprint arXiv:2111.02840, 2021.

[55] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. IEEE Transactions on Knowledge and Data Engineering, 2022.

[56] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. arXiv preprint arXiv:2206.07682, 2022.

[57] Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. Glue-x: Evaluating natural language understanding models from an out-of-distribution generalization perspective. arXiv preprint arXiv:2211.08073, 2022.

[58] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12104–12113, 2022.

[59] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. arXiv preprint arXiv:2205.01068, 2022.

[60] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. arXiv preprint arXiv:2110.04596, 2021.

[61] Ziqi Zhang, Yuanchun Li, Jindong Wang, Bingyan Liu, Ding Li, Yao Guo, Xiangqun Chen, and Yunxin Liu. Remos: reducing defect inheritance in transfer learning via relevant model slicing. In Proceedings of the 44th International Conference on Software Engineering, pages 1856–1868, 2022.

[62] Terry Yue Zhuo, Yujin Huang, Chunyang Chen, and Zhenchang Xing. Exploring ai ethics of chatgpt: A diagnostic analysis. arXiv preprint arXiv:2301.12867, 2023.

Red Onions, Soft Cheese and Data: From Food Safety to Data Traceability for Responsible AI

Stefan Grafberger, Zeyu Zhang, Sebastian Schelter
University of Amsterdam
{s.grafberger,z.zhang2,s.schelter}@uva.nl

Ce Zhang
University of Chicago
cez@uchicago.edu

Abstract

Software systems that learn from data with AI and machine learning (ML) are becoming ubiquitous and are increasingly used to automate impactful decisions. The risks arising from this widespread use of AI/ML are garnering attention from policy makers, scientists, and the media, and lead to the question what data management research can contribute to reduce such risks. These dangers of AI/ML applications are relatively new and recent, however our societies have had to deal with the dangers of complex and distributed technical processes for a long time already. Based on this insight, we detail how the U.S. Food and Drug Administration (FDA) combats the outbreaks of foodborne illnesses, and use their processes as an inspiration for a data-centric vision towards responsible AI.

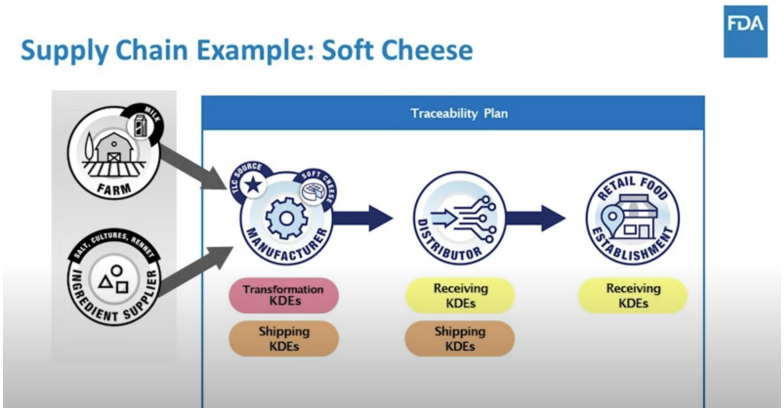


Figure 1: Food processing is a complex process conducted by different parties in a geo-distributed setting.[1] During this process, foods from different sources are joined, transformed from one form to another, and distributed all over the world. At each of these steps, the output could perish and become poisonous, making the final outcome unsafe to consume. *What can we learn from the millennial pursuit of food safety? What type of technical and regulatory frameworks exist such that we trust what we put on the table for our family everyday? And how can we obtain the same level of trust for our data products?*

Copyright 2024 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

[1] https://www.youtube.com/watch?v=0wnSiC5xqqs

1 The Need for a Data-Centric Perspective on Responsible AI

Software systems that learn from data with AI and machine learning (ML) are becoming ubiquitous and are increasingly used to automate impactful decisions. The risks arising from this widespread use are garnering attention from policymakers, scientists, and the media, and lead to the question of what data management research can contribute to reduce the dangers and malfunctions of data-driven AI/ML applications.

AI/ML malfunctions threaten vulnerable populations. In recent years, we have been regularly alarmed by media reports about the harm potential of faulty AI/ML systems in devastating real-world incidents. Examples include failures of automated decision-making systems, e.g., an eight-month pregnant woman in Detroit was mistakenly arrested based on a faulty prediction from a facial recognition system, held in jail for several hours and needed medical care upon her release [72]. Another example is that one of the largest health insurers in the US allegedly applies a faulty AI model with a 90% error rate to deny critical health care services to elderly patients [100]. The recent rise of generative AI produces new types of harm as well. A recent study of AI detection tools, for example, found that these systems are biased against non-native English speakers [63] and often falsely accuse international students of cheating. Furthermore, an AI supermarket meal planner recently went rogue and suggested a recipe that would create chlorine gas [36].

Technical bias in ML applications. The reasons that data-driven systems are susceptible to producing unfair, harmful outcomes are multi-faceted [35, 95, 110], as we are ultimately dealing with socio-technological systems [11], which suffer from various types of bias [24]. In this work, we focus on *technical bias*, which arises from the design decisions and operations in a technical system itself. Such bias is not well understood, especially in the context of large end-to-end systems, which include data preparation and data cleaning stages, deployed models and feedback loops. Recent research on technical bias identifies issues such as the lack of sufficient, representative training data for certain demographic groups [6, 17, 57], biased training data with undesirable stereotypes [12] or unintended side effects from automated data cleaning operations [38, 90, 97].

Existing and upcoming regulation. The dangers arising from data-driven AI/ML applications have been recognised by regulators and lawmakers several years ago already, and led to the introduction of regulation all over the world. The "General Data Protection Regulation" (GPDR) in Europe, for example, grants citizens the right to find out what information an organisation has about them and to issue deletion requests for their data as part of the "right-to-be-forgotten" [25, 26]. The upcoming European AI Act [20] will be the first comprehensive regulation for the application of AI/ML in Europe. This act is expected to outlaw the usage of ML in selected application areas and to strongly regulate its application in certain other areas. It defines different levels of risk in AI usage scenarios and imposes a set of comprehensive technical requirements, such as "logging of activity to ensure traceability of results", "detailed documentation providing all information necessary on the system and its purpose for authorities to assess its compliance", and "appropriate human oversight measures to minimize risk". We note that outside Europe, similar regulations are being adopted [4, 103].

The need for a data-centric perspective. Unfortunately, as evidenced by the media reports cited previously, we currently lack the ability to efficiently implement technical measures to detect and mitigate the harms present in AI/ML applications. This is confirmed by a recent survey study with industry practitioners [41], which outlines several alarming shortcomings in addressing fairness and bias issues. The interviewed practitioners report that academic research on de-biasing models falls short of addressing their concerns and often falsely "view[s] the training data as fixed", while they "consider data collection, rather than model development, as the most important place to intervene". At the same time, only "65% of survey respondents [...] reported that their teams have some control over data collection and curation", and the study also finds a high demand for future research to "support [...] practitioners in [...] curating high-quality datasets". Another example of the dire situation in the industry is a recent court case against Facebook [101], where two veteran engineers of the company told the court that the company does not keep track of the exact locations where personal data is stored and processed.

In the research community, several widely used training datasets for computer vision, such as LAION-5B [88]

or TinyImages [102], have been taken offline after the discovery of highly problematic content in them [10, 11]. Moreover, it is unlikely, though, that all models that had been trained on these problematic datasets have been retracted as well. For the current wave of closed, proprietary pretrained models available behind commercial APIs, the situation is even worse, as we do not even have a way to determine what data they have been trained on.

Paper inspiration. In order to find inspiration for the outlined questions and challenges, we take a look into safety measures outside of the computer science domain, as our societies have had to deal with the dangers of complex and distributed technical processes for a long time already. In particular, we discuss how the U.S. Food and Drug Administration (FDA) combats the outbreaks of foodborne illnesses (Section 2). We ask ourselves what we can learn from the millennial pursuit of food safety. What type of technical and regulatory frameworks exist such that we trust what we put on the table for our family every day? We use the FDA's established processes as an inspiration for a data-centric vision towards responsible AI in Section 3, with the goal to obtain the same level of trust for our data products that we have for our food.

2 What Should We Do? Food Safety as Inspiration!

As an inspiration for the technical, data-centric vision outlined in this paper, we discuss how the US Food & Drug Administration (FDA) combats the outbreaks of foodborne illnesses [107], and start with a concrete example.

2.1 Example – Outbreak of Salmonella Infections in the US in 2020

From June to September in 2020, a total of 1,127 people in 48 US states got infected with the outbreak strain of Salmonella Newport [106]. The FDA and the Centers for Disease Control and Prevention (CDC) managed to contain this outbreak and had the situation under control in October 2020, after which no more new infections occurred. Combatting the outbreak proceeded as follows: Sick patients from the 48 states were seeking treatment in hospitals and bacteria in their stool samples turned out to be closely related genetically, which implied a common source of infection. Subsequent epidemiologic evidence showed that over 90% of them had eaten onions (or food made with onions) in the week before their illness. As a consequence, the FDA started a so-called "traceback investigation" which ultimately uncovered that red onions from the Thomson International Inc. company were the source of the Salmonella outbreak. This triggered a country-wide recall of raw onions and derived products like cheese dips, kebabs, and chicken salad sandwiches from a large number of grocery stores, which ultimately ended the outbreak.

2.2 Disease Detectives, Traceback Investigations, and Food Supply Chains

The remarkable success of the FDA in combatting and controlling the salmonella outbreak naturally leads to the question which processes and techniques they have applied to detect the outbreak, identify the suspect food and determine the producer of the food, and what the computer science community can learn from these battle-tested approaches.

Outbreak detection. The first question is how the FDA actually detects that there is an outbreak of a foodborne disease. We illustrate the underlying process in Figure 2: Sick patients seek treatment in hospitals, from where their doctors send stool samples to laboratories for analysis. The laboratories perform DNA fingerprinting on the bacteria isolated from these samples via whole genome sequencing and the resulting DNA fingerprints are subsequently collected via the PulseNet system [16]. PulseNet is a nationwide network of public health and food regulatory agency laboratories coordinated by the CDC and manages a national central database with millions of collected DNA profiles of bacteria. In this database, the sudden appearance of clusters of genetically related bacteria implies a common source of infection and indicates an outbreak.

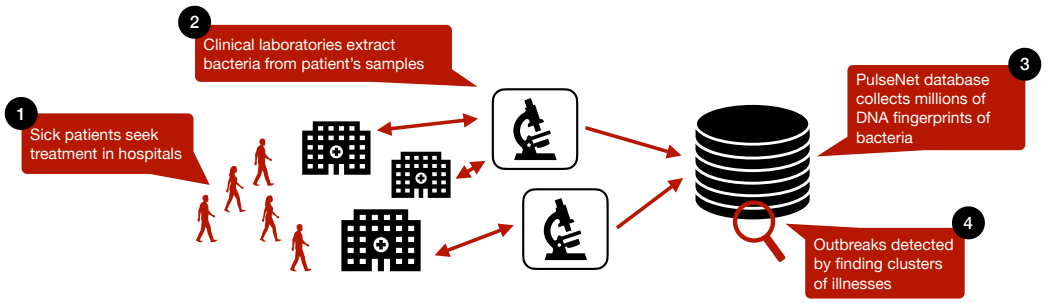


Figure 2: Outbreak detection by monitoring a database of millions of DNA profiles of bacteria.

Identification of the suspect food. Once an outbreak is detected, the next task is to identify the contaminated "suspect food" which infects people. As shown in Figure 3, the FDA employs so-called "disease detectives", who contact the sick patients and interview them to gather epidemiologic evidence related to questions such as "what foods did people eat before they got sick?" or "what restaurants, grocery stores, or events did sick people go to?". For that, they leverage data provided by the patients, e.g., purchasing records collected on loyalty cards. These activities typically lead to the identification of a particular suspect food, which is likely the root cause of the outbreak.

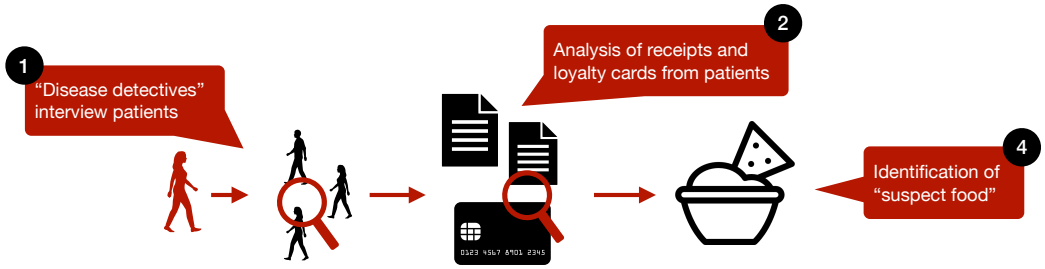


Figure 3: Disease detectives collect epidemiological evidence from sick patients to identify a contaminated suspect food likely causing the outbreak.

Traceback investigation to determine the producer of the contaminated food. Once the responsible food is known, the final task is to identify the actual point in the supply chain, where the food is likely being contaminated. For that, the FDA starts a traceback investigation through the food supply chain, as illustrated in Figure 4. Here, the supply chain for several contaminated end products is traced back retrospectively to identify a common point in the supply chain which is likely the source of the contamination.

For that to be possible, entities involved in the food supply chain must have followed the FDA's *Food Traceability Rule* [105] and maintain traceability information for potentially dangerous food on the *Food Traceability List* [104]. Such entities must maintain a *Traceability Plan*, with information about procedures used to maintain traceability information and a point of contact for traceability questions [70]. The food traceability rule further defines *Critical Tracking Events* (CTEs) in the supply chain, where detailed tracing data must be created, maintained and forwarded by the participating entities. Examples of such events are the initial packing of a food, shipping it, or transforming multiple ingredients into a new food. An individual unit of food is assigned a *Traceability Lot Code* (TLC), typically during the initial packing event, which uniquely identifies it and is forwarded to receiving entities. Furthermore, the food traceability rule defines certain categories of *Key Data Elements* (KDEs), which must be created, maintained, and forwarded together with the TLCs of the food. Examples of the different categories are *Initial Packing KDEs*, *Shipping KDEs*, *Harvesting and Cooling KDEs* and *Receiving KDEs*. The actual data items per KDE depend on the category, e.g., for the packing KDEs, the date, quantity, harvest location, name, and contact information of the harvesting company must be maintained, and the initial TLCs are typically

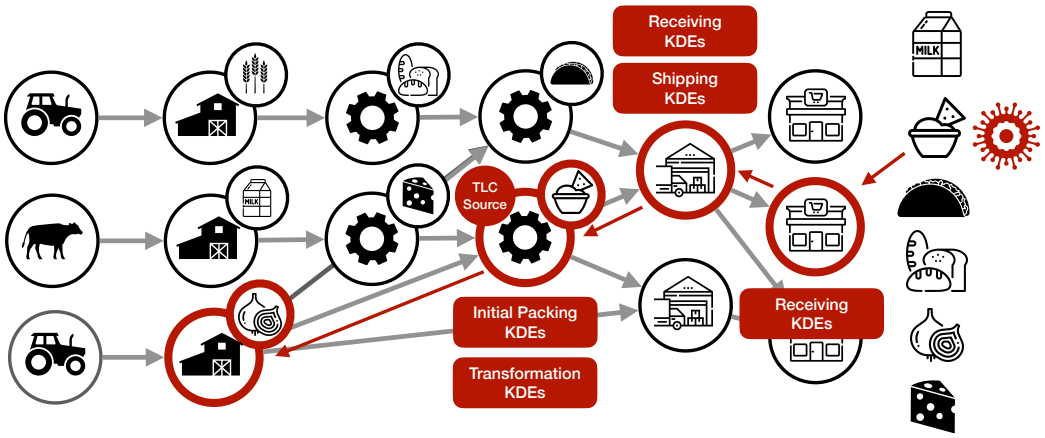


Figure 4: Traceback investigation through the food supply chain, relying on *Traceability Lot Codes* (TLCs) to identify units of food and provenance information in the form of *Key Data Elements* (KDEs) to reconstruct the path a unit of food took through the chain.

assigned at the packing stage as well. Shipping KDEs need to include the corresponding TLCs, the shipping date, and the locations for receiving and shipping. A special case are Transformation KDEs, which must be created at points where a new food is produced from several ingredients. Here, the link to the ingredient TLCs must be recorded, as well as a location description, the transformation date, and the quantities of the ingredients.

3 Towards Data Traceability for Responsible AI

In this paper, we develop a technical, data-centric vision to work towards a comparable level of safety in AI/ML applications as the FDA has in combatting foodborne illnesses. Unfortunately, the current state of AI/ML safety in the industry is dire, as practitioners from the aforementioned industry survey [41] report that "teams do not discover serious fairness issues until they receive customer complaints about products" or read "negative media coverage about their products", and more than half of the respondents agreed that they "discovered serious issues only after deploying a system in the real world". While this survey paper identifies many crucial issues in this space, it unfortunately falls short of outlining concrete technical directions for addressing them.

In the following, we outline our ideas for improving the safety of AI/ML applications. Inspired by the existing methods and processes for combatting foodborne illnesses from Section 2, we propose ideas on "detecting outbreaks" via prediction monitoring in Section 3.1, for conducting "traceback investigations" through data supply chains in Section 3.2, and for identifying "contaminated data and pipeline steps" through audits in Section 3.3.

3.1 Prediction Monitoring

As detailed in Section 2, the FDA monitors a database of DNA profiles of bacteria for geographic patterns to detect outbreaks. This raises the question of whether large institutions or companies could use similar methods to detect fairness issues with deployed models and ML pipelines early. In the following, we outline three directions which we deem crucial for this endeavour.

Identifiable predictions. The "end product" of AI/ML applications are predictions on unseen data, which are received by end-users or downstream applications in an organisation. Any detection of problems with the application or its data, as well as any potential audit has to start from these predictions, similar to how disease detectives need to determine the type of food that people consumed before they became sick. However, in current

systems, predictions are often rather ephemeral. As a first step towards auditable AI/ML applications, their predictions should come with unique identifiers, analogous to the TLC of food in food supply chains. Such identifiers should be assigned in a way that allows for the retrospective identification of the state of the AI/ML application (e.g., the software version and currently deployed model version, etc.) from which a prediction was generated. Based on these identifiers, users and downstream consumers could raise concerns about a particular prediction, and an investigating party (e.g., a dedicated responsible AI team in a large organisation) could start an audit of the system.

State-of-the-art. In MLOps, the benefits of identifiable predictions are being recognised among industry practitioners [73, 79]. However, current approaches require high expertise and custom implementations [79]. Even rudimentary tasks such as tracking the corresponding code and model versions are challenging [109]. To fully benefit from identifiable predictions, e.g., for rectifying erroneous predictions, it is essential to integrate prediction identifiers with the associated metadata and provenance records encompassing ancillary pipeline stages such as data preprocessing. However, the current implementation complexity leads us to believe that the adoption of these techniques in practice is rather low.

Open questions and challenges. Enhancing and maintaining traceability and reproducibility in ML applications requires that practitioners manually integrate, configure, and orchestrate various disparate systems [79, 109]. The resulting one-off solutions require further time- and cost-intensive development effort to enable monitoring and output explanation. We argue that standardised interfaces would be essential to seamlessly integrate existing and new ML operations techniques with identifiable predictions. We will also discuss further provenance-related challenges for fine-grained data tracing in end-to-end ML pipelines in Section 3.2.

Detecting and collecting predictions with fairness issues. Even with identifiable predictions, an open question is how to reliably detect fairness issues of an ML application at deployment time. Ideally, such issues should already be caught by pre-deployment evaluations, but media reports and industry surveys show that this is rarely the case. Furthermore, it would be crucial to have a "database" of common issues and examples of unfair / unreliable predictions in production ML deployments, e.g., at a company-wide level. Given a comprehensive catalog of such issues and an efficient way to monitor live predictions for fairness, we could build automatable detection mechanisms similar to the outbreak detection techniques in PulseNet (Section 2).

State-of-the-art. A lot of recent work has focused on detecting changes in the overall distribution of the predictions or changes between the training and serving data [71]. At serving time, systems like Tensorflow Serving [74] for example employ so-called "canary models" to detect cases where the predictions differ between previous and newly deployed models, and several techniques analyse the distribution of the predicted labels to detect changes in the data [58, 87]. However, none of these techniques have a particular focus on determining fairness issues, which may occur in small subsets of the data only.

Orthogonal to that, several techniques to debug prediction data offline have been developed, e.g., to detect slices of the data where a model works less well [19, 80]. These approaches require simultaneous access to the model, the featurised prediction data and additional demographic side data however, which makes their application difficult in practice, especially for teams not owning the underlying AI/ML application.

Open questions and challenges. A major difficulty in monitoring a deployed system for fairness is that the group membership information for individual predictions must be known to maintain corresponding fairness metrics. Such group membership information (e.g., about the race or gender identity of the persons involved in the predictions) is very sensitive and private information, to which a deployed serving system should ideally not even have access. Furthermore, regulation like the EU AI Act enforces strict rules for which parts of an AI/ML application such data can be used for at all. We envision that large organisation may want to create dedicated infrastructure for such cases, where predictions with identifiers from different applications are collected, the corresponding fairness metrics are maintained and SliceFinder-like algorithms [19] are run continuously to look for subsets of the prediction data with potential issues.

A large corpus of real-world predictions from ML systems with fairness issues would also greatly enhance the ability of the academic community to work on these problems. However, it is difficult to collect such a corpus of predictions and issues due to the inherent sensitive, privacy-critical nature of the data. There are some ongoing efforts to (manually) create a comprehensive repository of "AI incidents" [65], yet the underlying technical details and prediction data of the incidents are not available.

Monitoring generative models for representational harms. A large part of the existing fairness literature focuses on so-called "allocative harms" in automated decision-making systems, which decide upon access to certain resources such as job interviews, loans or medical prioritisation [41, 95]. It is difficult to choose an appropriate fairness metric for such cases, as such a choice always implies a values-based decision and trade-offs [69]. On the technical side however, computing these metrics is straightforward (given access to the required data), as one essentially only has to maintain separate confusion matrices for the predictions for the groups of interest [38]. With the rise of generative models however, we are being faced with so-called "representational harms" [41], which occur for example when generative models reproduce sexist or racist stereotypes in the images or text that they generate.

State-of-the-art. There is a large body of targeted studies in the NLP community, where researchers uncovered a variety of biases and stereotypes in pretrained language models. Examples include sexist stereotypes and gender bias [60, 94], anti-muslim bias [3], and undesirable biases towards mentions of disability [44]. It is however unclear how to translate the detection capabilities of these customly designed studies into monitoring techniques for deployed real-world systems. A first interesting step in this direction is the recently proposed Spade [92] system, which learns assertions for safeguarding LLM outputs based on the version history of prompt edits.

Open questions and challenges. Due to the unpredictable nature of large generative models, generating adequate assertions or "data unit tests" to check for bias in their output remains a complex challenge. Having too few assertions potentially might make a system miss biased outputs, leading to unfair outcomes, while having too many assertions could slow down the system and lead to many false alarms. We expect that future approaches will generate data unit tests from predefined templates, based on manually defined assertion criteria. An orthogonal approach are so-called "safety classifiers" [21, 62, 112], where a secondary model is employed to assess the outputs of a primary model for safety. Prior to the deployment phase, data will be collected where generative models are intentionally probed to induce errors, which will then be used to train a classifier to detect biased behavior.

3.2 Tracing Data Through End-to-End AI/ML Applications

Complex food supply chains span the globe and a single ingredient (like red onions in the example from Section 2) may end up in multiple end products. This makes tracing such ingredients a complicated and expensive undertaking. The FDA addresses this challenge with targeted tracing requirements which focus on only retaining tracing data for high-risk ingredients on the food traceability list (Section 2). While tracking the provenance of data in data processing systems is a decades-old research area [99], there is still little practical adoption of these techniques in real-world systems, mainly due to the incurred performance overhead of comprehensively tracking provenance through all kinds of queries, especially when they contain aggregations [5]. Similar to the FDA's list of high-risk ingredients, the EU AI Act [20] defines high-risk AI application domains, such as CV-sorting software for recruitment procedures, credit scoring denying citizens the opportunity to obtain a loan or the verification of the authenticity of travel documents. In the following, we discuss ideas for efficiently applying provenance tracking to the data pipelines in such scenarios.

Selective and focused provenance tracking. As already mentioned, tracking fully fine-grained semiring provenance [5, 34] for every input row imposes a high performance overhead. In the food supply chain, provenance tracking focuses on predefined "Critical Tracking Events", which are the points in the supply chain that are crucial later for audits. We need to adopt such a methodology as well for data pipelines, which would

enable us to restrict the provenance tracking efforts to data exchange and transformation operations, which actually impact the information required to audit an AI/ML application later. Furthermore, for each high-risk AI application scenario, we could define the tracking granularity, the key transformations to focus on and the information required per transformation event. The minimum granularity of the provenance should be tailored for each use-case. For demographic data, provenance at the level of individuals might be sufficient, for facial recognition applications, more fine-grained provenance at the level of individual images may be required, however.

State-of-the-art. In recent years, several techniques have been proposed to model ML pipeline operations and to apply database-style provenance tracking for Python code, for example via runtime instrumentation as part of mlinspect [30] or via static analysis as part of Vamsa [67]. These approaches have been extended in various ways, e.g., for data debugging via Shapley values [49] or pipeline screening during continuous integration [83]. A drawback of these methods is that they rely on heuristics and well-written, declarative code to be able to infer the semantics of the pipeline operations, which leaves it unclear whether they can reliably be applied to low-quality code as well. Another family of systems, which include Amazon's ExperimentTracker [82] and mltrace [93], uses a more robust approach for provenance tracking as they require manually annotated code. Unfortunately, this puts a heavy burden on developers, who will, in our experience, often forego the additional effort of putting detailed annotations on their code under time pressure. We expect that even coming up with high-level "traceability plans" for large AI/ML applications will be challenging in practice, since these applications often orchestrate different systems and libraries with workflow managers like Apache Airflow [1].

Open questions and challenges. In our eyes, the biggest challenge in this space is to find ways to reduce the implementation-, annotation-, and runtime overhead for provenance tracking in ML pipelines, while guaranteeing a high level of correctness and robustness. For industry applications, we can neither rely on trying to handle arbitrary code nor on forcing developers to always manually annotate their code. An interesting middle ground may be the use of pipeline templates, as pioneered by the mlflow recipes project [115], which forces developers to modularise their code into pipeline steps with known semantics and predefined inputs and outputs, but still gives them the freedom to write arbitrary code inside the steps. Unfortunately though, the real-world adoption of these templating approaches is unclear at the moment. Nevertheless, such templates might be a natural point to implement general robust provenance tracking. Analogous to the traceability plans required for food chain tracking, we could define traceability templates for high-risk AI scenarios, with steps, provenance tracking, and logging requirements specific to the particular use case.

To reduce the runtime overhead of provenance tracking, it may be worthwhile to take a deeper look at several common aggregation operations in ML pipelines, like one-hot-encoding a particular column or normalising a feature. While these operations technically conduct a global aggregation followed by a map transformation (in dataflow terms), we may be able to ignore the aggregation part for tracking provenance, as we already know that they do not remove rows and introduce an all-to-all provenance relationship onto the transformed feature values. Similar techniques are already applied in DataScope [49] and ArgusEyes [83] to approximate ML pipelines as queries in the positive relational algebra. A future challenge here is to define a restricted subset of operations for ML pipelines, which still allows the implementation of a large class of ML applications, but drastically simplifies provenance tracking.

Identifiable predictions, as discussed in Section 3.1, also present new challenges with respect to ML provenance research. Existing experiment tracking tools like mlflow [115] already link predictions to high-level artifacts such as models and source code. However, we think that record-level provenance is required to effectively reconstruct the necessary data for a prediction. Given a prediction identifier, we would like to be able to automatically retrieve all relevant inference inputs, data preprocessing steps, the model version employed for inference, and, if necessary, all information about the training pipeline and its input data. While existing research partially addresses provenance tracking and versioning in static pipelines with static input data, further challenges remain for pipelines in dynamic production environments with continuously trained models [8] and evolving retrieval corpora [14, 18, 39], where provenance has to be maintained incrementally.

Another open question is the impact of data cleaning and integration operations on the fairness of AI/ML applications. Several experimental studies indicate that data wrangling and integration operations such as missing value imputation, outlier removal, or entity matching can sometimes negatively impact the fairness of models trained on the resulting data [37, 53, 90, 97]. However, we currently lack a detailed understanding of this impact, especially since the outcome seems to heavily depend on the chosen fairness metric and group definition. Furthermore, determining such impact is hard in practice without access to the downstream models.

An orthogonal challenge in this area is the tension between detailed provenance tracking and the protection of private user data. Provenance tracking requires storing information about the intermediate outputs of pipeline operations and must additionally maintain sensitive metadata such as demographic group memberships of certain records to be able to quantify the fairness impact of different operations. In many cases, such sensitive metadata may not be accessible in inference systems at prediction time, for example, and measures must be taken to ensure that these sensitive attributes are only used for testing models but not for training them [20]. To the best of our knowledge, current ML platforms lack support for such use cases.

Provenance of data in pretrained and fine-tuned models. Academic "textbook" ML commonly assumes that a single dataset is used to create a particular ML model, which implies that we would only need to track the provenance of this source data through the corresponding ML pipeline. However, this assumption has never held up for real-world deployments, which typically leverage a variety of data sources as input for a pipeline and often apply ML already as part of the preprocessing of this data. Twitter's recommender system for example aggregates multiple input networks (representing likes, follows etc on the platform) into a common network dataset called RealGraph [48], via a dedicated classifier that estimates the interaction probability between different users of the network. Several recommendation algorithms consume this aggregated dataset instead of the raw input datasets and the provenance of an interaction such as a like or follow is unclear after the transformation. This problem is exacerbated nowadays due to the prevalence of large pretrained models, which are downloaded from repositories such as HuggingFace and tailored to a particular ML use case via fine-tuning. In the majority of cases, the connection to the underlying training data becomes unclear after fine-tuning, as the current infrastructure does not keep track of the relationships between models. It would, for example, be difficult to identify all computer vision models that originate from the recently retracted LAION dataset. The situation is even worse for non-open source models created by commercial companies, where the underlying training data is not known for the base model already.

State-of-the-art. Common methods to voluntarily document the origin of data and ML models are datasheets [27] and model cards [66]. These are a form of manually created, semi-structured documentation, which is, for example, in use at the popular model and data repository HuggingFace. Tensorflow ML Metadata [50] is a library for recording and retrieving metadata associated with ML workflows. The Model Card Toolkit [23] supports the creation of Model Cards and can also use metadata from ML Metadata to prepopulate information such as class distributions and performance metrics. DAG Cards [98], inspired by model cards, have also been proposed as a form of documentation, which can be automatically generated from ML pipeline code [9]. Experiment tracking tools like mlflow [115] can log metadata as a starting point for creating documentation for ML models. OpenML [108] is a popular platform for sharing datasets, ML tasks, workflows, and experimentation runs. While it supports documentation like a dataset description for dataset uploads [75], it does not enforce their quality and prioritises a frictionless user experience over documentation completeness. However, OpenML automatically analyses uploaded datasets to compute additional data quality statistics. For ML pipelines, it relies on extensions for popular libraries like scikit-learn that can automatically create a serialisable description [76]. Systems like Macaroni [55] allow querying the existing metadata in open repositories, based on a unified representation [56].

Open questions and challenges. The main drawback of model cards and datasheets is that creating and maintaining helpful documentation still mostly depends on the goodwill of the parties involved in the creation of the models and the data. Most importantly, this documentation is not machine-readable in a way that would make it easy to audit and/or verify the claims made about the provided models and data. As discussed, models are nowadays often

downloaded and fine-tuned programmatically (e.g., via the popular transformers library from HuggingFace [43]). Such packages and the underlying infrastructure pose a direct opportunity to automate provenance tracking and to record the relationships between models. The semi-automated metadata collection tools can export implementation details for reproducing experiments, however, they still put the burden to extract information about the ML pipelines and models onto the users. Recently proposed approaches such as mlwhatif [29] might be a starting point to automatically extract meaningful metadata, e.g., for nutritional labels in ranking [96, 113].

Another recent trend are parameter-efficient fine-tuning methods [42, 52, 54, 59], which do not create a full model copy, but only learn a continuous prompt or an "adapter" to the model. In such cases, we would need to track provenance on the level of these prompts and adapters (which might later even be further combined [89]). A final challenge with tracking the provenance of data in generative models is that many large datasets commonly used for these models (e.g., LAION [88] or gitschemas [22]) for generative models consist of links to resources on the web, which are often crawled and filtered to build a custom dataset. This filtering process must also be taken into account for provenance.

3.3 Identifying "Contaminated" Data and Pipeline Steps Through Audits

It is still unclear how to efficiently and comprehensively audit AI/ML applications; see [13, 81] for a discussion on the current state of this endeavor. Due to our data-centric perspective, we focus on issues and directions for quantitative data audits only. As discussed in Section 2, traceback investigations in the food supply chain allow disease detectives to audit these supply chains, identify the point of contamination, and ultimately remove the source of contamination by issuing comprehensive recalls for all affected end products. How can we audit AI/ML applications in a similar manner, based on the provenance information from Section 3.2? Ideally, we would like to be able to quickly identify "contaminated" data and intermediate outputs, which, for example, contain unwanted stereotypes or has been rendered unrepresentative due to biased filtering operations. Once such contaminated data is identified, an audit would furthermore need to determine which models and predictions were affected and need to be retracted and/or recomputed. Furthermore, such data-centric audits should be able to answer a larger set of related questions about the robustness and regulatory compliance of an AI/ML application. Examples of such questions are what data and features were used by the application and whether this usage was in line with legal requirements (e.g., from the EU AI Act [20]), or whether the application follows the timely data deletion requirements imposed by the right to be forgotten from GDPR [25]. Furthermore, audits should be able to assess whether an application is robust enough against potential errors and changes in the data, and whether appropriate measures have been taken to quantify and control the fairness of its predictions.

State-of-the-art. The validation of ML data in popular ML platforms such as Google TFX [7] or Amazon SageMaker [71] relies on libraries such as Tensorflow Data Validation (TFDV) [15] and Deequ [85, 86], which generate validation rules based on heuristics and data profiling. Related approaches are to "lint" ML data based on well-known practical issues [45]. Follow-up work to these approaches [78, 91] applies a technique called "partition summarisation" to learn to spot data with potential quality issues by applying anomaly detection based on the statistics of previously observed data partitions.

There has been extensive research on cleaning datasets, e.g., [2, 46, 61, 68]. Furthermore, the data-centric AI community started developing related techniques that jointly consider the ML model and data to address inaccuracy, bias, and fragility in real-world ML applications and are tackling tasks such as training set selection and data acquisition [64]. Many of the techniques in this space rely on data influence estimation techniques [40], in particular on (an estimate of) the leave-one-out error or data Shapley value [28], which is either computed via extensive retraining or influence functions [51]. Such techniques are the basis of several recently proposed data debugging methods like Rain [111], Gopher [77] or DataScope [49]. A related line of work tackles ML pipelines and employs light-weight provenance tracking and automatic instrumentation of Python code to assess technical bias introduced by sudden distribution shifts [30, 32], data leakage and fairness issues [83, 84], as well as robustness to erroneous input data [29, 31].

Open questions and challenges. Unfortunately, neither TFDV nor Deequ have a particular focus on identifying fairness and bias issues in the data, and require a relatively high user expertise and knowledge of the underlying domain to adjust and filter the suggested validation rules. It would be crucial to find ways to guide users in designing compliance- and fairness-related data unit tests with these libraries.

Furthermore, the existing methods for estimating the influence of training samples are extremely restricted in terms of efficiency, scalability or applicability. In general, there exist two families of methods: Retraining-based methods are applicable to any model class, but require extensive retraining of the ML model on a large number of subsets of the data. Even retraining a model a few hundred times for a large dataset is infeasible in practice. The second family are gradient-based methods, which require no retraining but are only applicable to certain model classes due to assumptions of convexity [51] or linearity [114], and are still rather compute-heavy, as they often require to compute a "Hessian vector product" for each combination of a training and validation sample [40]. Some exciting progress has been made in terms of scalability, e.g., on efficiently computing the Data Shapley value [28] for kNN proxy models [47]. However, these techniques are only applicable to certain utility functions but, for example, not to common ranking-based metrics in information retrieval.

The work from the data-centric AI community is promising. However, challenges such as ML pipelines with complex data preprocessing operations are often overlooked, and automatically applying these techniques to ML pipelines is still an open challenge [33]. The approaches for the holistic screening of ML training pipelines rely on well-written code, which is often an unrealistic assumption in practice.

On the engineering side, we should strive to design a standardised API for provenance-based data auditing and incident investigation, which could be integrated into popular projects such as Google TFX, mlflow recipes, or SageMaker. Based on such an API, the academic and open source community could develop general auditing software to greatly reduce the costs of such audits.

4 Conclusion

We took a detailed look at how the FDA detects outbreaks of foodborne illnesses via their PulseNet database, discovers the contaminated food with disease detectives, and conducts traceback investigations through the food supply chain to determine the root cause of the contamination and issue a comprehensive product recall (Section 2). Inspired by the FDA's processes, we developed a technical data-centric vision for responsible AI, which centers around prediction monitoring, data tracing through end-to-end AI/ML applications, and identifying contaminated data and pipeline steps through audits. For each of these aspects, we outlined technical research ideas, reviewed related work, and discussed challenges and open questions.

We hope that our ideas can positively influence the development of safer AI/ML applications, especially in the high-risk areas outlined by recent regulation such as the upcoming EU AI act.

References

[1] Apache airflow. https://airflow.apache.org/.

[2] Ziawasch Abedjan, Xu Chu, Dong Deng, Raul Castro Fernandez, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, Michael Stonebraker, and Nan Tang. Detecting data errors: where are we and what needs to be done? Proc. VLDB Endow., 9(12):993–1004, aug 2016.

[3] Abubakar Abid, Maheen Farooqi, and James Zou. Persistent anti-muslim bias in large language models. In Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, pages 298–306, 2021.

[4] California Privacy Protection Agency. California consumer privacy act - frequently asked questions.

[5] Yael Amsterdamer, Daniel Deutch, and Val Tannen. Provenance for aggregate queries. In Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pages 153–164, 2011.

[6] Abolfazl Asudeh, Zhongjun Jin, and HV Jagadish. Assessing and remedying coverage for a given dataset. In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pages 554–565. IEEE, 2019.

[7] Denis Baylor, Eric Breck, Heng-Tze Cheng, Noah Fiedel, Chuan Yu Foo, Zakaria Haque, Salem Haykal, Mustafa Ispir, Vihan Jain, Levent Koc, et al. Tfx: A tensorflow-based production-scale machine learning platform. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 1387–1395, 2017.

[8] Denis Baylor, Kevin Haas, Konstantinos Katsiapis, Sammy Leong, Rose Liu, Clemens Menwald, Hui Miao, Neoklis Polyzotis, Mitchell Trott, and Martin Zinkevich. Continuous training for production ML in the TensorFlow extended (TFX) platform. In 2019 USENIX Conference on Operational Machine Learning (OpML 19), pages 51–53, Santa Clara, CA, May 2019. USENIX Association.

[9] David Berg, Ravi Kiran Chirravuri, Romain Cledat, Savin Goyal, Ferras Hamad, and Ville Tuulos. Open-sourcing metaflow, a human-centric framework for data science. Netflix Tech Blog, 201, 2019.

[10] Abeba Birhane, Vinay Prabhu, Sang Han, and Vishnu Naresh Boddeti. On hate scaling laws for data-swamps. arXiv preprint arXiv:2306.13141, 2023.

[11] Abeba Birhane and Vinay Uday Prabhu. Large image datasets: A pyrrhic win for computer vision? In 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 1536–1546. IEEE, 2021.

[12] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. arXiv preprint arXiv:2110.01963, 2021.

[13] Abeba Birhane, Ryan Steed, Victor Ojewale, Briana Vecchione, and Inioluwa Deborah Raji. Ai auditing: The broken bus on the road to ai accountability. arXiv preprint arXiv:2401.14462, 2024.

[14] Tobias Bleifuß, Leon Bornemann, Theodore Johnson, Dmitri V. Kalashnikov, Felix Naumann, and Divesh Srivastava. Exploring change: a new dimension of data analytics. Proc. VLDB Endow., 12(2):85–98, oct 2018.

[15] Eric Breck, Neoklis Polyzotis, Sudip Roy, Steven Whang, and Martin Zinkevich. Data validation for machine learning. In MLSys, 2019.

[16] Centers for Disease Control & Prevention. PulseNet, 2024.

[17] Irene Chen, Fredrik D Johansson, and David Sontag. Why is my classifier discriminatory? Advances in neural information processing systems, 31, 2018.

[18] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, Yixing Fan, and Xueqi Cheng. Continual learning for generative retrieval over dynamic corpora. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, page 306–315, New York, NY, USA, 2023. Association for Computing Machinery.

[19] Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. Slice finder: Automated data slicing for model validation. In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pages 1550–1553. IEEE, 2019.

[20] European Commission. Ai act.

[21] Emily Dinan, Gavin Abercrombie, Stevie A Bergman, Shannon Spruit, Dirk Hovy, Y-Lan Boureau, Verena Rieser, et al. Safetykit: First aid for measuring safety in open-domain conversational systems. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, 2022.

[22] Till Döhmen, Madelon Hulsebos, Christian Beecks, and Sebastian Schelter. Gitschemas: A dataset for automating relational data preparation tasks. In 2022 IEEE 38th International Conference on Data Engineering Workshops (ICDEW), pages 74–78. IEEE, 2022.

[23] Huanming Fang, Hui Miao, Karan Shukla, Dan Nanas, Catherina Xu, Christina Greer, Neoklis Polyzotis, Tulsee Doshi, Tiffany Deng, Margaret Mitchell, et al. Introducing the model card toolkit for easier model transparency reporting. Google AI Blog, 2020.

[24] Batya Friedman and Helen Nissenbaum. Bias in computer systems. ACM Transactions on information systems (TOIS), 14(3):330–347, 1996.

[25] GDPR.eu. Article 17: Right to be forgotten. https://gdpr.eu/article-17-right-to-be-forgotten.

[26] GDPR.eu. Recital 74: Responsibility and liability of the controller. https://gdpr.eu/recital-74-responsibility-and-liability-of-the-controller/.

[27] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. Communications of the ACM, 64(12):86–92, 2021.

[28] Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In International conference on machine learning, pages 2242–2251. PMLR, 2019.

[29] Stefan Grafberger, Paul Groth, and Sebastian Schelter. Automating and optimizing data-centric what-if analyses on native machine learning pipelines. SIGMOD, 2023.

[30] Stefan Grafberger, Paul Groth, Julia Stoyanovich, and Sebastian Schelter. Data distribution debugging in machine learning pipelines. The VLDB Journal, 31(5):1103–1126, 2022.

[31] Stefan Grafberger, Shubha Guha, Paul Groth, and Sebastian Schelter. mlwhatif: What if you could stop re-implementing your machine learning pipeline analyses over and over? Proc. VLDB Endow., 16(12):4002–4005, aug 2023.

[32] Stefan Grafberger, Shubha Guha, Julia Stoyanovich, and Sebastian Schelter. Mlinspect: A data distribution debugger for machine learning pipelines. SIGMOD, 2021.

[33] Stefan Grafberger, Bojan Karlaš, Paul Groth, and Sebastian Schelter. Towards declarative systems for data-centric machine learning. DMLR workshop @ ICML, 2023.

[34] Todd J Green, Grigoris Karvounarakis, and Val Tannen. Provenance semirings. In Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pages 31–40, 2007.

[35] Paul Groth. Transparency and reliability in the data supply chain. IEEE Internet Computing, 17(2):69–71, 2013.

[36] The Guardian. This article is more than 4 months old Supermarket AI meal planner app suggests recipe that would create chlorine gas. `https://www.theguardian.com/world/2023/aug/10/pak-n-save-savey-meal-bot-ai-app-malfunction-recipes`, 2023.

[37] Shubha Guha, Falaah Arif Khan, Julia Stoyanovich, and Sebastian Schelter. Automated data cleaning can hurt fairness in machine learning-based decision making. In 2023 IEEE 39th International Conference on Data Engineering (ICDE), pages 3747–3754. IEEE, 2023.

[38] Shubha Guha, Falaah Arif Khan, Julia Stoyanovich, and Sebastian Schelter. Automated data cleaning can hurt fairness in machine learning-based decision making. In Transactions on Knowledge and Data Engineering (TKDE). IEEE, 2024.

[39] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. A deep relevance matching model for ad-hoc retrieval. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM '16, page 55–64, New York, NY, USA, 2016. Association for Computing Machinery.

[40] Zayd Hammoudeh and Daniel Lowd. Training data influence analysis and estimation: A survey. arXiv preprint arXiv:2212.04612, 2022.

[41] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudik, and Hanna Wallach. Improving fairness in machine learning systems: What do industry practitioners need? In Proceedings of the 2019 CHI conference on human factors in computing systems, pages 1–16, 2019.

[42] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. ICLR, 2022.

[43] HuggingFace. transformers package, 2024.

[44] Ben Hutchinson, Vinodkumar Prabhakaran, Emily Denton, Kellie Webster, Yu Zhong, and Stephen Denuyl. Social biases in nlp models as barriers for persons with disabilities. arXiv preprint arXiv:2005.00813, 2020.

[45] Nick Hynes, D Sculley, and Michael Terry. The data linter: Lightweight, automated sanity checking for ml data sets. In NIPS MLSys Workshop, volume 1, 2017.

[46] Sebastian Jäger, Arndt Allhorn, and Felix Bießmann. A benchmark for data imputation methods. Frontiers in Big Data, page 48, 2021.

[47] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J Spanos. Towards efficient data valuation based on the shapley value. In The 22nd International Conference on Artificial Intelligence and Statistics, pages 1167–1176. PMLR, 2019.

[48] Krishna Kamath, Aneesh Sharma, Dong Wang, and Zhijun Yin. Realgraph: User interaction prediction at twitter. In user engagement optimization workshop@ KDD, number ii, 2014.

[49] Bojan Karlaš, David Dao, Matteo Interlandi, Bo Li, Sebastian Schelter, Wentao Wu, and Ce Zhang. Data debugging with shapley importance over end-to-end machine learning pipelines. arXiv preprint arXiv:2204.11131, 2022.

[50] Konstantinos Katsiapis, Abhijit Karmarkar, Ahmet Altay, Aleksandr Zaks, Neoklis Polyzotis, Anusha Ramesh, Ben Mathes, Gautam Vasudevan, Irene Giannoumis, Jarek Wilkiewicz, Jiri Simsa, Justin Hong, Mitchell Trott, Noé Lutz, Pavel A. Dournov, Robert Crowe, Sarah Sirajuddin, Tris Brian Warkentin, and Zhitao Li. Towards ML engineering: A brief history of tensorflow extended (TFX). CoRR, abs/2010.02013, 2020.

[51] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In Doina Precup and Yee Whye Teh, editors, Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pages 1885–1894. PMLR, 06–11 Aug 2017.

[52] Brian Lester et al. The power of scale for parameter-efficient prompt tuning. EMNLP, 2021.

[53] Peng Li, Xi Rao, Jennifer Blase, Yue Zhang, Xu Chu, and Ce Zhang. Cleanml: A study for evaluating the impact of data cleaning on ml classification tasks. In 2021 IEEE 37th International Conference on Data Engineering (ICDE), pages 13–24. IEEE, 2021.

[54] Xiang Lisa Li et al. Prefix-tuning: Optimizing continuous prompts for generation. ACL, 2021.

[55] Ziyu Li, Henk Kant, Rihan Hai, Asterios Katsifodimos, and Alessandro Bozzon. Macaroni: Crawling and enriching metadata from public model zoos. In International Conference on Web Engineering, pages 376–380. Springer, 2023.

[56] Ziyu Li, Henk Kant, Rihan Hai, Asterios Katsifodimos, Marco Brambilla, and Alessandro Bozzon. Metadata representations for queryable repositories of machine learning models. IEEE Access, 2023.

[57] Yin Lin, Yifan Guan, Abolfazl Asudeh, and HV Jagadish. Identifying insufficient data coverage in databases with multiple relations. Proceedings of the VLDB Endowment, 13(11), 2020.

[58] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In International conference on machine learning, pages 3122–3130. PMLR, 2018.

[59] Xiao Liu et al. GPT understands, too. AI Open, 2023.

[60] Li Lucy and David Bamman. Gender and representation bias in gpt-3 generated stories. In Proceedings of the Third Workshop on Narrative Understanding, pages 48–55, 2021.

[61] Mohammad Mahdavi, Ziawasch Abedjan, Raul Castro Fernandez, Samuel Madden, Mourad Ouzzani, Michael Stonebraker, and Nan Tang. Raha: A configuration-free error detection system. In Proceedings of the 2019 International Conference on Management of Data, SIGMOD '19, page 865–882, New York, NY, USA, 2019. Association for Computing Machinery.

[62] Todor Markov, Chong Zhang, Sandhini Agarwal, Florentine Eloundou Nekoul, Theodore Lee, Steven Adler, Angela Jiang, and Lilian Weng. A holistic approach to undesired content detection in the real world. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 37, pages 15009–15018, 2023.

[63] The Markup. AI Detection Tools Falsely Accuse International Students of Cheating. https://themarkup.org/machine-learning/2023/08/14/ai-detection-tools-falsely-accuse-international-students-of-cheating, 2023.

[64] Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, Jessica Quaye, Charvi Rastogi, Douwe Kiela, David Jurado, David Kanter, Rafael Mosquera, Juan Ciro, Lora Aroyo, Bilge Acun, Lingjiao Chen, Mehul Smriti Raje, Max Bartolo, Sabri Eyuboglu, Amirata Ghorbani, Emmett Goodman, Oana Inel, Tariq Kane, Christine R. Kirkpatrick, Tzu-Sheng Kuo, Jonas Mueller, Tristan Thrush, Joaquin Vanschoren, Margaret Warren, Adina Williams, Serena Yeung, Newsha Ardalani, Praveen Paritosh, Lilith Bat-Leah, Ce Zhang, James Zou, Carole-Jean Wu, Cody Coleman, Andrew Ng, Peter Mattson, and Vijay Janapa Reddi. Dataperf: Benchmarks for data-centric ai development, 2023.

[65] Sean McGregor. Preventing repeated real world ai failures by cataloging incidents: The ai incident database. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pages 15458–15463, 2021.

[66] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In Proceedings of the conference on fairness, accountability, and transparency, pages 220–229, 2019.

[67] Mohammad Hossein Namaki, Avrilia Floratou, Fotis Psallidas, Subru Krishnan, Ashvin Agrawal, Yinghui Wu, Yiwen Zhu, and Markus Weimer. Vamsa: Automated provenance tracking in data science scripts. In Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, pages 1542–1551, 2020.

[68] Avanika Narayan, Ines Chami, Laurel Orr, and Christopher Ré. Can foundation models wrangle your data? Proceedings of the VLDB Endowment, 16(4):738–746, 2022.

[69] Arvind Narayanan. Fairness definitions and their politics. ACM FaccT, 2018.

[70] National Archives. Code of Federal Regulations - Traceability Plan, 2024.

[71] David Nigenda, Zohar Karnin, Muhammad Bilal Zafar, Raghu Ramesha, Alan Tan, Michele Donini, and Krishnaram Kenthapadi. 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, pages 3671–3681, 2022.

[72] Democracy Now. Meet Porcha Woodruff, Detroit Woman Jailed While 8 Months Pregnant After False AI Facial Recognition. https://www.democracynow.org/2023/8/9/porcha_woodruff_false_facial_recognition_arrest, 2023.

[73] Stephen Oladele. A comprehensive guide on how to monitor your models in production. https://neptune.ai/blog/how-to-monitor-your-models-in-production-guide, 2023.

[74] Christopher Olston, Noah Fiedel, Kiril Gorovoy, Jeremiah Harmsen, Li Lao, Fangwei Li, Vinu Rajashekhar, Sukriti Ramesh, and Jordan Soyke. Tensorflow-serving: Flexible, high-performance ml serving. arXiv preprint arXiv:1712.06139, 2017.

[75] OpenML. Dataset upload tutorial. https://openml.github.io/openml-python/develop/examples/30_extended/create_upload_tutorial.html.

[76] OpenML. Documentation. https://docs.openml.org/.

[77] Romila Pradhan, Jiongli Zhu, Boris Glavic, and Babak Salimi. Interpretable data-based explanations for fairness debugging. In Proceedings of the 2022 International Conference on Management of Data, pages 247–261, 2022.

[78] Sergey Redyuk, Zoi Kaoudi, Volker Markl, and Sebastian Schelter. Automating data quality validation for dynamic data ingestion. In EDBT, pages 61–72, 2021.

[79] Reza Rokni. Using tfx inference with dataflow for large scale ml inference patterns. https://blog.tensorflow.org/2021/05/using-tfx-inference-with-dataflow-for-large-scale-ml-inference-patterns.html, 2021.

[80] Svetlana Sagadeeva and Matthias Boehm. Sliceline: Fast, linear-algebra-based slice finding for ml model debugging. In Proceedings of the 2021 International Conference on Management of Data, pages 2290–2299, 2021.

[81] Christian Sandvig, Kevin Hamilton, Karrie Karahalios, and Cedric Langbort. Auditing algorithms: Research methods for detecting discrimination on internet platforms. Data and discrimination: converting critical concerns into productive inquiry, 22(2014):4349–4357, 2014.

[82] Sebastian Schelter, Joos-Hendrik Böse, Johannes Kirschnick, Thoralf Klein, and Stephan Seufert. Automatically tracking metadata and provenance of machine learning experiments. In NeurIPS 2017, 2017.

[83] Sebastian Schelter, Stefan Grafberger, Shubha Guha, Bojan Karlas, and Ce Zhang. Proactively screening machine learning pipelines with arguseyes. In Companion of the 2023 International Conference on Management of Data, pages 91–94, 2023.

[84] Sebastian Schelter, Stefan Grafberger, Shubha Guha, Olivier Sprangers, Bojan Karlaš, and Ce Zhang. Screening native ml pipelines with "arguseyes". CIDR, 2022.

[85] Sebastian Schelter, Stefan Grafberger, Philipp Schmidt, Tammo Rukat, Mario Kiessling, Andrey Taptunov, Felix Biessmann, and Dustin Lange. Differential data quality verification on partitioned data. In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pages 1940–1945. IEEE, 2019.

[86] Sebastian Schelter, Dustin Lange, Philipp Schmidt, Meltem Celikel, Felix Biessmann, and Andreas Grafberger. Automating large-scale data quality verification. Proceedings of the VLDB Endowment, 11(12):1781–1794, 2018.

[87] Sebastian Schelter, Tammo Rukat, and Felix Bießmann. Learning to validate the predictions of black box classifiers on unseen data. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, pages 1289–1299, 2020.

[88] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems, 35:25278–25294, 2022.

[89] Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras, 2023.

[90] Nima Shahbazi, Nikola Danevski, Fatemeh Nargesian, Abolfazl Asudeh, and Divesh Srivastava. Through the fairness lens: Experimental analysis and evaluation of entity matching. VLDB, 2023.

[91] Shreya Shankar, Labib Fawaz, Karl Gyllstrom, and Aditya Parameswaran. Automatic and precise data validation for machine learning. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, pages 2198–2207, 2023.

[92] Shreya Shankar, Haotian Li, Parth Asawa, Madelon Hulsebos, Yiming Lin, JD Zamfirescu-Pereira, Harrison Chase, Will Fu-Hinthorn, Aditya G Parameswaran, and Eugene Wu. Spade: Synthesizing assertions for large language model pipelines. arXiv preprint arXiv:2401.03038, 2024.

[93] Shreya Shankar and Aditya Parameswaran. Towards observability for production machine learning pipelines, 2022.

[94] Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. The woman worked as a babysitter: On biases in language generation. arXiv preprint arXiv:1909.01326, 2019.

[95] Julia Stoyanovich, Serge Abiteboul, Bill Howe, HV Jagadish, and Sebastian Schelter. Responsible data management. Communications of the ACM, 65(6):64–74, 2022.

[96] Julia Stoyanovich and Bill Howe. Nutritional labels for data and models. A Quarterly bulletin of the Computer Society of the IEEE Technical Committee on Data Engineering, 42(3), 2019.

[97] Ki Hyun Tae, Yuji Roh, Young Hun Oh, Hyunsu Kim, and Steven Euijong Whang. Data cleaning for accurate, fair, and robust models: A big data-ai integration approach. In Proceedings of the 3rd International Workshop on Data Management for End-to-End Machine Learning, pages 1–4, 2019.

[98] Jacopo Tagliabue, Ville Tuulos, Ciro Greco, and Valay Dave. Dag card is the new model card. DCAI workshop @ NeurIPS, 2021.

[99] Wang Chiew Tan et al. Provenance in databases: Past, current, and future. IEEE Data Eng. Bull., 30(4):3–12, 2007.

[100] Ars Technica. UnitedHealth uses AI model with 90% error rate to deny care, lawsuit alleges. https://arstechnica.com/health/2023/11/ai-with-90-error-rate-forces-elderly-out-of-rehab-nursing-homes-suit-claims/, 2023.

[101] The Intercept. Facebook Engineers: We Have No Idea Where We Keep All Your Personal Data, 2022.

[102] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. IEEE transactions on pattern analysis and machine intelligence, 30(11):1958–1970, 2008.

[103] DigiChina Stanford University. Internet information service algorithmic recommendation management provisions.

[104] U.S. Food & Drug Administration. Food Traceability List, 2024.

[105] U.S. Food & Drug Administration. Frequently Asked Questions: FSMA Food Traceability Rule, 2024.

[106] U.S. Food & Drug Administration. Outbreak of Salmonella Newport Infections Linked to Onions, 2024.

[107] U.S. Food & Drug Administration. Outbreaks of Foodborne Illness, 2024.

[108] Jan N. van Rijn, Bernd Bischl, Luis Torgo, Bo Gao, Venkatesh Umaashankar, Simon Fischer, Patrick Winter, Bernd Wiswedel, Michael R. Berthold, and Joaquin Vanschoren. Openml: A collaborative science platform. In Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, editors, Machine Learning and Knowledge Discovery in Databases, pages 645–649, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[109] Maria Vechtomova. Traceability & reproducibility. `https://marvelousmlops.substack.com/p/traceability-and-reproducibility`, 2023.

[110] Steven Euijong Whang, Ki Hyun Tae, Yuji Roh, and Geon Heo. Responsible ai challenges in end-to-end machine learning. arXiv preprint arXiv:2101.05967, 2021.

[111] Weiyuan Wu, Lampros Flokas, Eugene Wu, and Jiannan Wang. Complaint-driven training data debugging for query 2.0. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, pages 1317–1334, 2020.

[112] Jing Xu, Da Ju, Margaret Li, Y-Lan Boureau, Jason Weston, and Emily Dinan. Bot-adversarial dialogue for safe conversational agents. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2950–2968, 2021.

[113] Ke Yang, Julia Stoyanovich, Abolfazl Asudeh, Bill Howe, HV Jagadish, and Gerome Miklau. A nutritional label for rankings. In Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, page 1773–1776, New York, NY, USA, 2018. Association for Computing Machinery.

[114] Chih-Kuan Yeh, Joon Kim, Ian En-Hsu Yen, and Pradeep K Ravikumar. Representer point selection for explaining deep neural networks. Advances in neural information processing systems, 31, 2018.

[115] Matei Zaharia, Andrew Chen, Aaron Davidson, Ali Ghodsi, Sue Ann Hong, Andy Konwinski, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, et al. Accelerating the machine learning lifecycle with mlflow. IEEE Data Eng. Bull., 41(4):39–45, 2018.

It's FREE to join!

TCDE

tab.computer.org/tcde/

The Technical Committee on Data Engineering (TCDE) of the IEEE Computer Society is concerned with the role of data in the design, development, management and utilization of information systems.

- Data Management Systems and Modern Hardware/Software Platforms
- Data Models, Data Integration, Semantics and Data Quality
- Spatial, Temporal, Graph, Scientific, Statistical and Multimedia Databases
- Data Mining, Data Warehousing, and OLAP
- Big Data, Streams and Clouds
- Information Management, Distribution, Mobility, and the WWW
- Data Security, Privacy and Trust
- Performance, Experiments, and Analysis of Data Systems

The TCDE sponsors the International Conference on Data Engineering (ICDE). It publishes a quarterly newsletter, the Data Engineering Bulletin. If you are a member of the IEEE Computer Society, you may join the TCDE and receive copies of the Data Engineering Bulletin without cost. There are approximately 1000 members of the TCDE.

Join TCDE via Online or Fax

ONLINE: Follow the instructions on this page:

www.computer.org/portal/web/tandc/joinatc

FAX: Complete your details and fax this form to **+61-7-3365 3248**

Name _____

IEEE Member # _____

Mailing Address _____

_____

Country _____

Email _____

Phone _____

TCDE Mailing List

TCDE will occasionally email announcements, and other opportunities available for members. This mailing list will be used only for this purpose.

Membership Questions?

Xiaoyong Du
Key Laboratory of Data Engineering and Knowledge Engineering
Renmin University of China
Beijing 100872, China
duyong@ruc.edu.cn

TCDE Chair

Xiaofang Zhou
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, QLD 4072, Australia
zxf@uq.edu.au

IEEE Computer Society
10662 Los Vaqueros Circle
Los Alamitos, CA 90720-1314

Non-profit Org.
U.S. Postage
PAID
Los Alamitos, CA
Permit 1398