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.

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

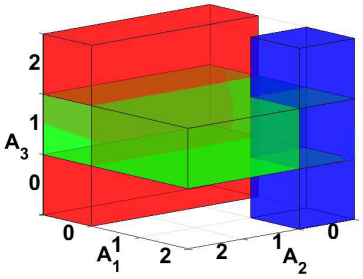

Figure 1: Categorical attributes: the uncovered region of a toy example, as the collection of three MUPs.

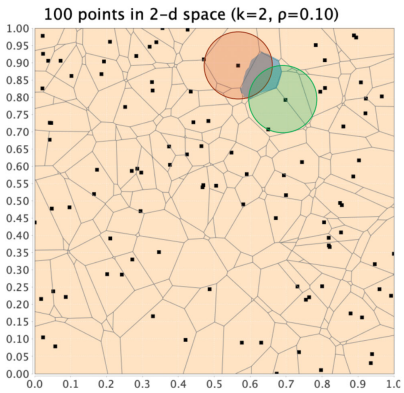

Figure 2: Continuous attributes, 2D: identifying the covered region in the gray Voronoi cell.

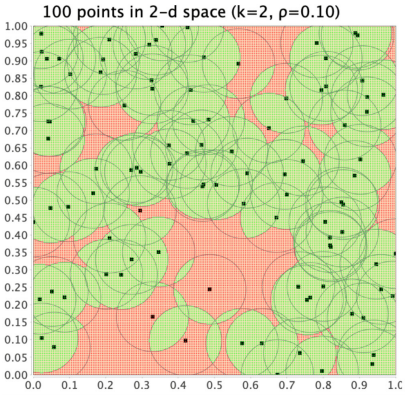

Figure 3: Continuous attributes, 2D: Uncovered region marked in red.

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. $q \in \cap_{\forall t \in S} \circ_t$, 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.

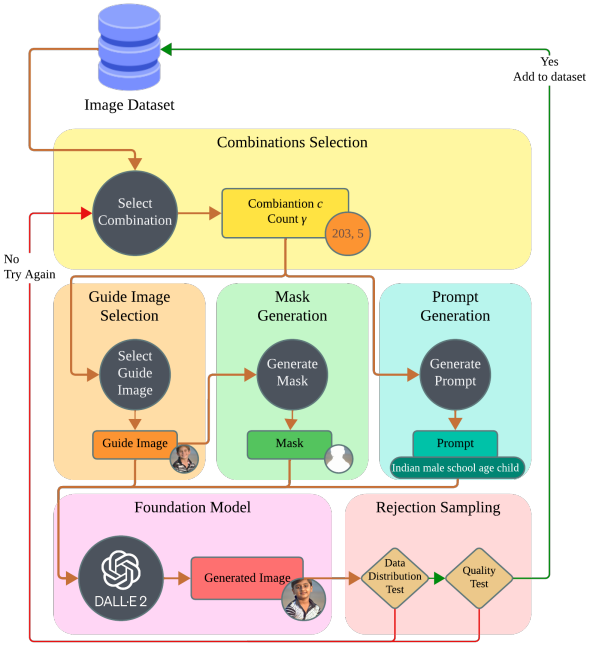

Figure 5: Architecture of CHAMELEON for image data augmentation for coverage enhancement.

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

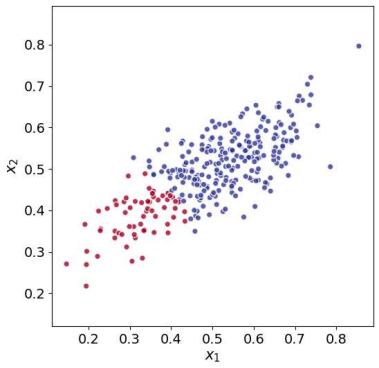

Figure 6: Data set $\mathcal{D}$ generated using a Gaussian distribution; x_1 and x_2 are positively correlated

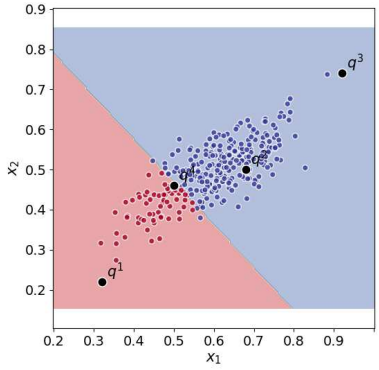

Figure 7: The decision boundary of learned model h and query points $\mathbf{q}^1$ to $\mathbf{q}^4$

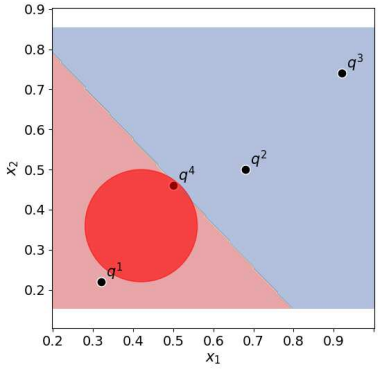

Figure 8: Ground-truth boundary, overlaid on the model decision boundary and query points

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

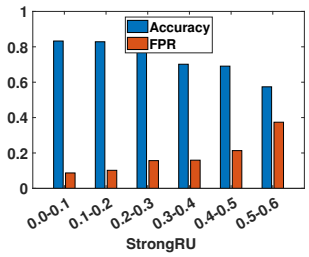

Figure 9: *Adult*, efficacy of STRONGRU on classification

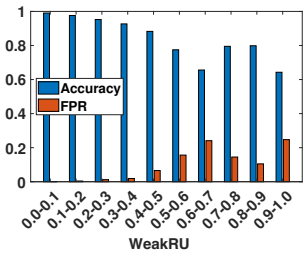

Figure 10: *Adult*, efficacy of WEAKRU on classification

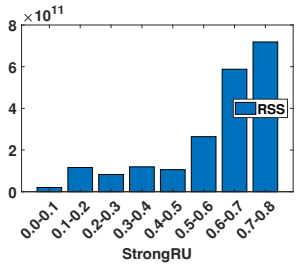

Figure 11: *House Sales in King County*, efficacy of STRONGRU on regression

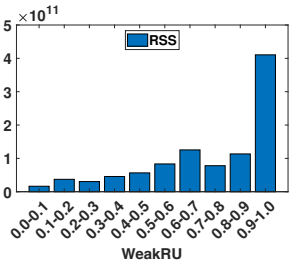

Figure 12: *House Sales in King County*, efficacy WEAKRU on regression

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.