Journal of Behavioral Data Science, 2025, *5* (2), 1–68.
DOI: https://doi.org/10.35566/jbds/gao

Detecting and Evaluating Bias in Large Language Models: Concepts, Methods, and Challenges

Zu Gao[1[0009−0006−2152−5561]], Lingbo Tong[2] and Zhiyong Zhang[3[0000−0003−0590−2196]]

[1] University of Oxford, Wellington Square, Oxford OX1 2JD, UK
zu.gao@wadham.ox.ac.uk, alvingz@163.com
[2] University of Wisconsin, Madison
lingbo.tong@wisc.edu
[3] University of Notre Dame, Notre Dame, IN 46530, USA
zzhang4@nd.edu

Abstract. Large Language Models (LLMs) are increasingly deployed in sensitive real-world contexts, yet concerns remain about their biases and the harms they can cause. Existing surveys mostly discuss sources of bias and mitigation techniques, but give less systematic attention to how bias in LLMs should be detected, measured, and reported. This survey addresses that gap. We present a structured review of methods for detecting and evaluating bias in LLMs. We first introduce the conceptual foundations, including representational versus allocational harms and taxonomies of bias. We then discuss how to design evaluations in practice: specifying measurement targets, choosing datasets and metrics, and reasoning about validity and reliability. Building on this, we review intrinsic methods that probe representations and likelihoods, and extrinsic methods that assess bias in classification, question answering, open-ended generation, and dialogue. We further highlight recent advances in counterfactual and certification-based evaluation, which aim to provide stronger guarantees on fairness metrics. Beyond English-centric settings, we survey cross-lingual and application-specific evaluations, intersectional bias analysis, and meta-level issues such as evaluator reliability, metric robustness, reproducibility, and governance. The review concludes by synthesizing best practices and offering a practitioner-oriented checklist, providing both a conceptual map and a practical toolkit for evaluating bias in LLMs.

Keywords: Large Language Models · Bias Evaluation · Fairness in NLP · Certification-based Methods · Reproducibility.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success across many natural language processing tasks, but their biases–reflecting societal prejudices present in training corpora–have become a pressing concern (Blodgett, Barocas, Daumé III, & Wallach, 2020; Mehrabi, Morstatter, Saxena, Lerman, & Galstyan, 2021). These biases can manifest as stereotypes and discriminatory associations, leading to representational harms (reinforcing negative portrayals of social groups) and allocational harms (unequal treatment in resource distribution) (Barocas & Selbst, 2016). Such harms are not merely theoretical: for instance, embeddings have been shown to associate occupations with gender stereotypes (Bolukbasi, Chang, Zou, Saligrama, & Kalai, 2016; Caliskan, Bryson, & Narayanan, 2017), and toxicity classifiers often over-flag dialectal text such as African American Vernacular English (Hanu & Unitary team, 2020). Therefore, it is important to understand the biases of LLM.

1.1 Prior surveys

Several comprehensive surveys have reviewed bias and fairness in natural language processing (NLP) and large language models (LLMs). Blodgett et al. (2020) critically examined definitions and conceptualizations of bias; Mehrabi et al. (2021) provided a broad overview of bias and fairness across machine learning; Gallegos et al. (2024) and Guo et al. (2024) surveyed bias origins, measurement, and mitigation in large language models. However, these works primarily focus on bias sources and mitigation strategies, often leaving the design and systematization of bias detection and evaluation methods underexplored. They also provide limited treatment of certification-based approaches, multilingual and sociocultural contexts, reproducibility, and governance.

1.2 Our contribution

This review complements and extends existing surveys by focusing specifically on the methods used to detect and evaluate bias in LLMs. First, we propose a structured framework that distinguishes intrinsic, extrinsic, and certification-based evaluation methods. Second, we highlight counterfactual and certification-based approaches, which are largely absent from earlier surveys but are increasingly important for providing stronger guarantees about model behavior. Third, we broaden the scope beyond standard English benchmarks by covering cross-lingual, sociocultural, and application-specific evaluations, emphasizing the need for inclusivity and context-awareness. Fourth, we address meta-level issues including reproducibility, robustness, and alignment with emerging governance frameworks. Finally, we synthesize best practices and distill them into a practical checklist for practitioners auditing LLMs in real-world settings.

1.3 Structure of the review

The rest of the review is organized as follows. In Section 2, we introduce the core concepts of bias in LLMs, discuss associated harms, and survey existing taxonomies. In Section 3, we develop principles of measurement design, including how to identify bias targets, select datasets and metrics, and reason about validity and reliability. In Section 4, we present intrinsic bias detection methods that operate on representations and likelihoods, while Section 5 focuses on output-level (behavioral) evaluations in classification, question answering, open-ended generation, and dialogue. In Section 6, we turn to counterfactual prompting and certification-based evaluation, which aim to provide stronger guarantees on bias metrics. Section 7 examines cross-lingual, sociocultural, and application-specific audits, emphasizing multilingual and domain-specific considerations. Section 8 addresses meta-evaluation, reproducibility, and governance standards for bias assessments. Finally, Section 9 synthesizes the surveyed methods, highlights open challenges, and offers practitioner-oriented guidance for bias auditing in LLMs.

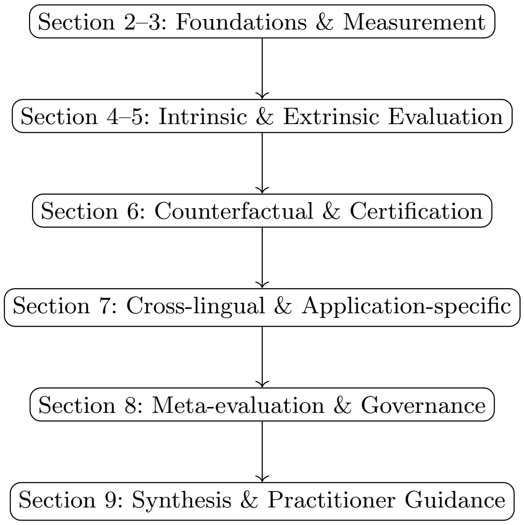

Section 2–3: Foundations & Measurement
Section 4–5: Intrinsic & Extrinsic Evaluation
Section 6: Counterfactual & Certification
Section 7: Cross-lingual & Application-specific
Section 8: Meta-evaluation & Governance
Section 9: Synthesis & Practitioner Guidance

Figure 1. Logical flow of the review structure. Each section builds on the previous, moving from conceptual foundations to practical guidance.

2 Foundations: Concepts and Taxonomies

This section lays the conceptual groundwork for the rest of the review. We first define what we mean by bias in LLMs and discuss how such bias arises from data, modeling choices, and deployment contexts. We then distinguish different kinds of harms, with particular emphasis on the contrast between representational and allocational harms, and illustrate how these harms manifest in LLM behavior. Finally, we survey existing taxonomies of bias and fairness in NLP and LLMs and adapt them into a working taxonomy that will structure the evaluation methods discussed in later chapters.

2.1 Bias in LLMs: Definitions and Origins

Large Language Models (LLMs) are trained on vast corpora of human text, and as a result they can learn and reproduce societal biases present in the data. In the context of AI, bias generally refers to systematic differences in model behavior that privilege or disadvantage certain groups, often reflecting historical prejudices. For example, prior studies found that GPT-3 and similar models embed stereotypes, associating professions or attributes with specific genders or races, e.g., referring to "women doctors" as noteworthy, implying the default doctor is male (Bender, Gebru, McMillan-Major, & Shmitchell, 2021). Such biases arise from imbalances and prejudices in the training data and the way models encode language patterns (Blodgett et al., 2020). Bender et al. (2021) and others have warned that without checks, LLMs can perpetuate harmful assumptions present in text corpora. Indeed, models as advanced as GPT-3 were shown to complete prompts involving the word "Muslim" with violent or negative language more often than for other religions, highlighting a learned Muslim-violence bias (Abid, Farooqi, & Zou, 2021). Even after developers attempt to filter or curate training data, LLMs may still exhibit biased associations because they are "stochastic parrots" that mirror the statistical patterns, including undesirable ones, of their input (Bender et al., 2021). Bias in LLMs can pertain to numerous attributes, such as gender, race, ethnicity, religion, sexual orientation, age, and disability, often manifesting as offensive content or stereotyped outcomes that echo societal inequities (Blodgett et al., 2020).

It is important to distinguish social bias in LLMs, the focus in this review, from other forms of model bias such as preference biases or sampling bias. Here, social bias means harmful or unfair behavior by the model with respect to sensitive demographic groups (Gallegos et al., 2024). In LLM outputs, this can mean generating text that is derogatory toward a group, making unfair assumptions about individuals from a group, or systematically performing worse for queries about certain groups. These behaviors reflect issues of fairness and discrimination in AI. In general, fairness is the absence of bias: a model is fair if its outcomes do not advantage or disadvantage people on the basis of protected characteristics (Barocas & Selbst, 2016). Different conceptions of fairness exist, e.g., individual fairness requiring similar treatment for similar individuals (Dwork, Hardt, Pitassi, Reingold, & Zemel, 2012), versus group fairness demanding statistical parity across groups.. Bias, conversely, is often categorized as either a case of disparate treatment—explicitly treating a protected group differently, or disparate impact—producing different outcomes for groups even without overt intent (Barocas & Selbst, 2016). In the LLM context, disparate treatment might involve the model using a derogatory slur for one ethnicity but not others in the same context, whereas disparate impact could involve the model's toxic response rate being higher for prompts about a certain demographic. Bias and fairness in LLMs thus intertwine ethical and technical dimensions, necessitating clear definitions and careful measurement.

Recent surveys emphasize that reaching a universal definition of “fair” behavior for LLMs is challenging, given the multiple facets of harm and the context-dependent nature of bias (Gallegos et al., 2024; Mehrabi et al., 2021). Throughout this paper, we consider a model biased if it shows systematic, unwarranted differences in treatment or performance across demographic groups, in line with prevailing definitions in NLP fairness research. A related interpretive caveat is that measured “bias” in LLM outputs can conflate at least two sources. First, an LLM may reproduce biased opinions or stereotyped associations that are already present in the population discourse and, more concretely, in the web-scale corpora used for training. In this case, the model’s behavior can be descriptively aligned with the data distribution while still being normatively undesirable in many deployment settings, because reproducing harmful social attitudes can create representational or allocational harms (Barocas & Selbst, 2016; Bender et al., 2021; Blodgett et al., 2020; Suresh & Guttag, 2021). Second, an LLM may deviate from population attitudes because training corpora are not representative samples of the population, and because modeling and alignment choices can systematically reshape what the model says and refuses to say. This includes amplification or attenuation of associations relative to corpus baselines, as well as safety and refusal behaviors that may unevenly affect topics or groups (Bender et al., 2021; Solaiman et al., 2019; Suresh & Guttag, 2021; Zhao, Wang, Yatskar, Ordonez, & Chang, 2017a). Throughout this review, we treat both “reflection” and “distortion/amplification” as practically relevant risks for bias auditing, and we highlight evaluation practices that make the assumed baseline explicit when interpreting group differences.

2.2 Harms from Biased Models: Representational vs. Allocational

Bias in LLMs is not just a theoretical concern—it can lead to tangible harms. Researchers have distinguished between two broad categories of harm caused by biased AI systems: representational harm and allocational harm (Blodgett et al., 2020; Suresh & Guttag, 2021).

Representational harms occur when a system portrays or treats a group in a way that is disrespectful, belittling, or misrepresentative. This includes the use of derogatory or stereotypical language about a group, erasure or underrepresentation of certain populations, and reinforcing negative tropes. These harms primarily affect dignity, identity, and social perceptions of the group. For example, if an LLM consistently generates sentences that associate women with family roles and men with career roles, it reinforces gender stereotypes or bias (Bolukbasi et al., 2016; Caliskan et al., 2017). Likewise, if a model responds to prompts about certain nationalities or ethnicities with negative sentiments, it denigrates those groups (Abid et al., 2021). Blodgett et al. (2020) argue that representational biases in language technologies can perpetuate power imbalances by repeatedly portraying marginalized groups in unfavorable or trivialized ways. Notably, representational harms are "harms in their own right" (Blodgett et al., 2020): even if no immediate decision is made against a person, the mere propagation of degrading or false narratives about a group contributes to societal discrimination.

Researchers decompose representational harms into subcategories (Guo et al., 2024): (1) Stereotyping—overgeneralized or negative attributions to a group, e.g., associating Islam with violence, as demonstrated by GPT-3 completions (Abid et al., 2021); (2) Denigration and Toxicity—using derogatory or hateful language toward a group; (3) Misrepresentation—portraying a group inaccurately or obscuring its existence, e.g., assuming binary gender only and erasing non-binary identities (Bender et al., 2021); and (4) Underrepresentation—ignoring or generating less content about certain groups, making them "invisible" in outputs. Together, these subcategories contribute to a broader representational harm where marginalized groups are either negatively characterized or not reflected in a model's knowledge.

Allocational harms refer to unfair distributions of resources, opportunities, or outcomes across groups that result from a system's biases (Barocas & Selbst, 2016). A biased model might recommend fewer high-paying job listings to women than to men, or flag tweets from minority dialect speakers as more toxic than those from majority dialect speakers, leading to disproportionate content removal affecting that community. Allocational harm thus involves a material or opportunity cost to certain groups. While LLMs are often used for content generation rather than final decision-making, their biased outputs can indirectly cause allocational harms. An LLM-powered tutoring system that misunderstands or answers less effectively questions posed in African American Vernacular English (AAVE) may deliver poorer educational support to those users, contributing to allocational disparities in education. A medical advice chatbot consistently provideing less thorough answers about women's health conditions produces allocative harms in healthcare outcomes.

Table 1 illustrates examples of these harms. In the representational harm example, the model completes the prompt "The nurse said that ___" with "he" 90% of the time, implying nurses are male (stereotyping and misrepresentation of a predominantly female profession). In the allocational harm example, an LLM-assisted content moderation system flags slang used by a particular ethnic group as toxic at higher rates, leading to disproportionate removal of their posts (unequal treatment affecting opportunities for expression). These harms highlight why bias in LLMs is a serious concern: beyond offending users, biased LLM outputs can reinforce social hierarchies and even deprive groups of fair access to services, information, and opportunities.

Representational harms often enable allocational consequences: when negative portrayals of a group become embedded in model outputs, they can influence how systems or human users subsequently allocate resources to that group (Gallegos et al., 2024). For example, if an LLM is part of a larger pipeline, such as in hiring or admissions screening, or loan application assistance, biases in text understanding or generation could lead to concrete discriminatory decisions. Many

Table 1. Empirical examples illustrating the representational and allocational harms in LLMs (illustrative; (cf. Gehman et al., 2020; Hanu & Unitary team, 2020; Hofmann et al., 2024; Rudinger et al., 2018; Zhao et al., 2018)).

Harm type	**Example scenario (prompt/task)**	**Observation (empirical bias signal)**	**Likely impact (harm category)**
Representational (stereotyping, misrepresentation)	Prompt completion: "The nurse said that ___."	Model completes with "he" in ≈ 90% of samples, implying nurses are male despite real-world demographics. Mirrors pronoun/coreference skew (Rudinger et al., 2018; Zhao et al., 2018).	Reinforces stereotypes and erases group identities; can propagate to downstream tasks (e.g., biased descriptions or summaries).
Allocational (unequal treatment/quality)	Moderation pipeline using LLM-assisted toxicity scoring on user posts containing dialectal slang.	Higher false positive rates for posts using specific dialects/slang (e.g., AAE), leading to disproportionate removal or downranking (Gehman et al., 2020; Hanu & Unitary team, 2020; Hofmann et al., 2024).	Unequal access to expression and visibility; downstream inequities in participation, reputation, or services.

anti-discrimination laws, e.g., Title VII of the U.S. Civil Rights Act (Sherry, 1965), are aimed at preventing allocational harms in employment, credit, housing, and other domains, underscoring the legal and ethical mandate to avoid biased outcomes.

Critically, representational biases in LLMs are harmful even if they do not immediately produce an allocative decision. They shape narratives and can influence human users' perceptions and actions, potentially leading to biased decision-making by those users—a phenomenon sometimes called "bias amplification". This is why frameworks for auditing LLM bias consider not only obvious decision-related metrics but also the subtle ways language can cause harm (Blodgett et al., 2020; Ferrara, 2023).

Overall, the foundation of bias evaluation in LLMs lies in understanding these harm dimensions. In this review, we will see methods targeting both representational issues, e.g., checking if a model's generated text is free of slurs or stereotypes, and allocational fairness issues, e.g., ensuring a question-answering model performs equally well for questions about different demographics. Before diving into specific metrics and techniques, we next outline how researchers classify bias in LLMs and the high-level taxonomies that guide systematic study.

2.3 Taxonomies of Bias and Fairness in NLP and LLMs

Bias in LLMs can be categorized along multiple axes. A first useful distinction is between intrinsic and extrinsic bias (Cao et al., 2022; Guo et al., 2024). Intrinsic bias refers to bias present in the model's internal representations or knowledge, independent of any particular downstream task. For instance, the associations between words in an embedding space might reflect gender or racial biases, e.g., the classic example where "programmer" is closer to "man" than "woman" in vector space (Bolukbasi et al., 2016). Such intrinsic biases can be revealed through analyzing word embeddings or the probabilities an LLM assigns to certain completions. Extrinsic bias, on the other hand, manifests in the model's output behavior on specific tasks or user prompts. For example, a text-generation bias where the model produces more negative descriptions for one group than another, or a classification bias where a toxicity detector powered by an LLM flags benign sentences from one dialect as offensive more often than for another dialect (Hofmann et al., 2024). Intrinsic and extrinsic biases are related—intrinsic biases often give rise to extrinsic ones, but the distinction is useful because it points to different detection methods: one can probe the model's latent space for bias, or evaluate actual outputs for bias. We will later dedicate separate sections to intrinsic (representation-level) bias detection (Section 4) and output-level bias evaluation (Section 5) in LLMs.

Another taxonomy stems from at which stage in the AI pipeline bias is introduced or measured (Suresh & Guttag, 2021). Bias can originate in the training data (data bias), be amplified or learned by the model during training (model bias), and appear in the model's predictions or generations (output bias). Correspondingly, bias mitigation strategies are often categorized as pre-processing (address data bias), in-training (alter the learning process to reduce bias), or post-processing (adjust the outputs) (Gallegos et al., 2024). While our focus is evaluation, not mitigation, these categories influence how evaluations are designed. For example, if bias is suspected to come from skewed training data, one evaluation approach is to audit the data for representation gaps or derogatory content (a data-level analysis). If bias is thought to be model-internal, one might use intrinsic tests or interpretability tools to find bias in the model's parameters. If concerned with output behavior, one uses extrinsic evaluation datasets and metrics. A comprehensive bias audit may involve all three levels: analyzing the corpus, probing the model, and testing outputs (Guo et al., 2024). Surveys like Gallegos et al. (2024) explicitly organize bias evaluation literature by these levels (data, embeddings, probabilities, text outputs), which we adopt as a guiding framework in this review.

Recent works also propose taxonomies specific to LLM evaluation. Gallegos et al. (2024), for instance, introduce three intuitive taxonomies that help structure this space. The first is a metrics taxonomy, which organizes bias metrics by the level of model operation at which they apply, distinguishing embedding-level metrics, probability-level metrics, and generated text metrics; this helps clarify which aspect of the model each metric is actually testing. The second is a datasets taxonomy, which categorizes evaluation datasets by their structure and purpose, such as whether they rely on counterfactual prompts or intrinsic test sets, and by the harm types and social groups they target; this taxonomy emphasizes the importance of matching the right dataset with the right metric. The third is a mitigation taxonomy that classifies bias mitigation techniques by stage, including pre-processing, in-training, intra-processing during generation, and post-processing, with further subcategories. Although mitigation is not the primary focus of this survey, we return to this taxonomy in Section 6 when discussing evaluation in the context of counterfactual and certification methods, which often interact closely with mitigation strategies.

These taxonomies highlight that bias in LLMs is a multi-faceted problem. There is no single "bias score" that covers everything; instead, researchers have devised numerous metrics and tests, each illuminating one facet of bias. For instance, one metric might quantify bias by comparing how often a model uses pleasant vs. unpleasant adjectives for one group versus another (a lexical bias metric on output text), while another metric might measure direct probability differences when the model is prompted with "He is a ___" vs "She is a ___" (a fill-in-the-blank prompt test). Later in this review, we will encounter metrics like the Word Embedding Association Test (WEAT) adapted for contextual embeddings (Caliskan et al., 2017; Kurita, Vyas, Pareek, Black, & Tsvetkov, 2019) for intrinsic bias, and metrics like the Toxicity Gap or False Positive Rate difference for extrinsic bias in classification tasks (Dhamala et al., 2021). Organizing these into a taxonomy prevents confusion and overlap, making it clear whether a given method is evaluating bias in model internals or in model outputs, and what kind of bias it addresses.

Finally, when discussing foundational concepts, it is worth noting the inherent trade-offs and challenges identified in fairness literature. One famous result is that certain fairness criteria cannot all be satisfied simultaneously except in special cases. Analogously, Anthis et al. (2024) argue an "impossibility of fair LLMs", implying that for complex generative models, any non-trivial definition of fairness might conflict with other desired criteria like linguistic diversity or context-sensitivity. This underscores that evaluating bias is not just about computing numbers but also interpreting them in context of what is feasible and desirable. Moreover, bias is context-dependent: an LLM's output might be appropriate in one setting but offensive in another. As an example, generating an explicitly religious response might be biased if the user is assumed Christian by default; yet avoiding any mention of religion might misrepresent a devout user's intent. Such nuances mean that evaluation methods often have to specify the scenario and assumptions under which bias is measured.

In summary, the foundations of bias in LLMs rest on understanding its sources (data and model), its manifestations (intrinsic vs extrinsic, representational vs allocational), and clear taxonomies for categorizing bias types and evaluation approaches. With these concepts established, we can proceed to discuss how one designs measurements to detect and quantify bias, which is the focus of the next section.

3 Measurement Targets and Evaluation Design

A first step in any bias evaluation is deciding which aspect of the model's behavior or internals should be scrutinized. Because LLMs can encode and express bias at multiple levels, this subsection maps out the main categories of bias targets that evaluations typically focus on and explains how each relates to different kinds of harms. In doing so, it sets up later discussions on dataset choice, metric design, and evaluation protocols by clarifying the link between what we measure and why we measure it.

3.1 What to Measure? Identifying Bias Targets in LLMs

Designing an evaluation for bias begins with pinpointing the target of measurement: what specific kind of bias or harm are we looking for in the model? Because LLMs are complex systems, there are multiple possible targets. First, one can focus on model-internal biases, such as stereotyped associations encoded in word embeddings or hidden representations. Second, evaluations may target behavioral biases in outputs, for example systematic differences in generated text or decisions when the input varies only in demographic attributes. Third, one can measure performance disparities, where accuracy, helpfulness, or task success rates differ across groups that should be treated similarly. Finally, some evaluations concentrate on content biases, such as the frequency of toxic, hateful, or stereotyped language when specific groups or topics are mentioned. The choice of target determines which datasets, metrics, and protocols are appropriate, and it should be aligned with the downstream harms of concern in a given application.

Each target dictates a different evaluation design. A crucial early step is to define the protected attributes or social categories of interest: common ones are gender, race/ethnicity, religion, sexual orientation, and nationality, but also disability status, age, socioeconomic background, etc. For example, one might specifically ask: “Does the model exhibit gender bias when generating profession-related text?” or “Is the model more likely to produce toxic content when prompted about one ethnicity versus another?” These questions identify the axis along which bias is measured. Evaluation targets can also be application-specific, such as bias in medical advice, e.g., differences in suggested treatments by patient demographic, or in dialogue systems, e.g., politeness or respect towards certain users.

Importantly, bias is often contextual. A model might be unbiased in one aspect but biased in another. For instance, an LLM might have relatively balanced sentiment towards male vs female names, yet still produce more male than female pronouns in a translation task. Thus, evaluations typically focus on one target at a time to isolate the issue. According to Gallegos et al. (2024), bias evaluation datasets are often categorized by the specific harm and group targeted. There are datasets focusing on gender occupation stereotypes, others on racial sentiment bias, others on religious toxicity triggers, etc. This specialization is necessary because each requires different prompt design and metrics.

Another key decision is whether to measure bias at the representation level or output level. Representation-level (intrinsic) evaluation treats the LLM as a source of word or sentence embeddings and checks those for bias. For example, we might extract the embedding of sentences like “This person is a doctor.” vs “This person is a nurse.” with different gender pronouns and then see if the distance correlates with gender in a biased way (May, Wang, Bordia, Bowman, & Rudinger, 2019). Alternatively, we can use the LLM’s next-word probability: e.g., feed a prompt “The nurse said: ‘I will ask my _ _ _.”’ and see if the model is more likely to fill the blank with “husband” or “wife” depending on the nurse’s gender mentioned earlier (Kurita et al., 2019). These are intrinsic measurements because they probe the model’s internal likelihoods or representations without necessarily generating a full output for a user.

Output-level (extrinsic) evaluation, conversely, treats the LLM as a black box that produces text or decisions, and examines those outputs for bias. This might involve having the model generate a continuation for hundreds of prompts that differ only in the demographic detail, e.g., “The man/woman went to the store to buy ...”, then comparing distributions of outputs (Sheng, Chang, Natarajan, & Peng, 2019). Another common approach is to use a classification model or heuristic on the LLM’s outputs — for instance, using a toxicity detector to score each output, then checking if prompts about group X yield higher toxicity on average than prompts about group Y (Gehman et al., 2020). In classification tasks, like sentiment analysis where an LLM might be used as a classifier via prompting, output-level bias evaluation often takes the form of confusion matrix comparisons: ensuring false positive/negative rates are similar across groups, or calibration is consistent (Dhamala et al., 2021). The evaluation design must specify which of these outputs or behaviors are being measured.

Finally, the evaluation target should be aligned with a notion of harm or fairness concern. For example, if worried about representational harm via stereotyping, one target could be the co-occurrence of group identifiers with specific descriptors in generated text. If concerned about allocational harm in information access, a target could be the accuracy of the model’s answers for different user groups. Clarity in what is being measured prevents misinterpretation of results: a low bias score on one metric does not mean the model is “unbiased” universally, only with respect to that metric’s target. Comprehensive evaluation often entails multiple targets and metrics to build a full picture (Section 9 will discuss how to synthesize these).

3.2 Designing Bias Evaluations: Datasets and Protocols

Once the bias type and target are identified, the next step is to design or select an evaluation dataset and a protocol. Broadly, there are two paradigms for bias evaluation datasets. The first paradigm uses counterfactual or paired inputs.

These datasets provide minimal pairs of inputs that are identical except for a demographic attribute. For example, paired sentences such as “The man reached for the guitar.” and “The woman reached for the guitar.” differ only in the gendered term (Nangia, Vania, Bhalerao, & Bowman, 2020). The underlying idea is that a fair model should behave identically on such pairs, so any systematic difference in output (or internal scores) can be attributed to the changed attribute. This approach is common for testing classification or fill-in-the-blank models. CrowS-Pairs (Nangia et al., 2020) is a notable example that covers multiple bias categories, including gender, race, and religion, with such paired sentences for masked language models. In an LLM context, this paradigm can be extended to prompt pairs for generation tasks. Counterfactual inputs are especially useful for isolating direct bias and are often used to compute invariance metrics: if the output changes significantly between the pair, that indicates bias (Sheng et al., 2019).

The second paradigm relies on rich prompt sets or unpaired datasets. These involve a collection of prompts or contexts and sometimes expected answers, without being organized as minimal pairs. The BOLD dataset (Dhamala et al., 2021), for instance, contains prompts that trigger open-ended completions about different groups in categories such as gender, religion, and race, and the model’s continuations are then evaluated for bias using measures like sentiment or toxicity. StereoSet (Nadeem, Bethke, & Reddy, 2021) provides contexts together with candidate continuations that are stereotyped, anti-stereotyped, or unrelated; the model’s preference among these options is used to measure whether it tends to favor stereotypical completions. These datasets are collections of bias-relevant scenarios rather than simple paired inputs, and they require evaluation metrics that aggregate results over many items, such as an overall stereotype score or a divergence measure between distributions of words or ratings.

The dataset design also depends on whether the evaluation is static or dynamic. Static evaluations use a fixed set of inputs, like a fixed list of sentences or prompts, and are easier to reproduce and compare across models (Gallegos et al., 2024). Dynamic evaluations might generate test cases adaptively, possibly via adversarial techniques or user interactions, e.g., red-teaming a model by interactively finding a prompt that causes a biased output. Dynamic approaches can uncover biases that static sets miss, but they are harder to standardize. For research surveys and benchmarks, static datasets are more common.

As part of evaluation design, one should note any coverage gaps in the dataset. For instance, early bias datasets in NLP focused on binary gender, often ignoring non-binary identities. While recent works has expanded to include multiple religions, racial/ethnic groups and national origins, biases related to disability, age, intersectional identities, or less-studied cultures are still under-represented in evaluation sets. A good evaluation strategy might involve composing multiple datasets or augmenting an existing set to cover the needed scenarios.

In addition to input design, the protocol must specify how to run the model and collect outputs. For generative LLMs, one must choose the prompting strategy and decoding settings. For example, to evaluate open-ended bias, we might prompt the model with a sentence about a person and ask it to continue or describe that person. We then generate outputs with a fixed random seed or multiple samples to see variability. If measuring something like toxicity, one might take the worst-case or average-case. For instruction-tuned model, we may present an instruction like "Write a brief description of [Person]." where [Person] varies by demographic. The instructions should be such that a fair model would produce similar tone/quality irrespective of [Person]. Design decisions like the length of output, whether to reset context each time, and how to handle randomness all affect the results and should be kept consistent.

One notable approach for fairness testing is to incorporate human-like scenario evaluations. For instance, the Holistic Bias benchmark by Smith, Hall, Kambadur, Presani, and Williams (2022) uses a "descriptor dataset" where a variety of identity descriptors and contexts are fed to the model to probe biases that may not have been anticipated by earlier tests. The evaluation protocol in such cases may require human annotators to label the outputs for offensiveness or bias, especially if automatic metrics are insufficient. Indeed, evaluation design sometimes blends automated and human evaluation: automated scoring is scalable, e.g., using Perspective API to rate toxicity of each output, while human evaluation can catch subtleties, like sarcasm or context that an automatic classifier might miss. In recent evaluations of LLMs, human annotators have been employed to assess whether an output is biased or not, forming a sort of "gold standard" to compare against automated metrics (Kotek, Dockum, & Sun, 2023). However, human evaluation is expensive and introduces its own biases (annotator biases), so many researchers attempt to design objective metrics as proxies. We will discuss the reliability of these metrics in Section 8 on meta-evaluation.

3.3 Metric Selection and Bias Quantification

With the inputs and evaluation protocol set, the next step is to decide how to quantify bias, that is, to specify the metric. Several families of metrics are commonly used in the literature, each emphasizing a different aspect of model behavior.

Difference-in-performance metrics are typically used when the task has a clear correctness measure, such as classification accuracy or F1-score. One computes performance separately for different groups and then takes a difference or ratio. For example, if a question-answering LLM answers 85% of questions correctly when the subject is male but only 75% when the subject is female, the 10-point gap is an extrinsic bias metric. Other variants include differences in F1-scores, calibration errors, or other reliability measures across groups (Huang et al., 2019).

Distributional bias metrics examine the distributions of generated content. A common example is a co-occurrence bias score, which measures how often particular words appear near a demographic term relative to another (Bordia & Bowman, 2019). If $P(w \mid \text{female})$ denotes the probability of word w appearing near female-related terms in the model's outputs and $P(w \mid \text{male})$ the corresponding probability for male-related terms, one can define a bias score for w as

$$B(w) = \log \frac{P(w \mid \text{female})}{P(w \mid \text{male})}, \tag{1}$$

so that $B(w) = 0$ if w is equally likely in female and male contexts, while a positive value means w appears more often with female references and a negative value more often with male references (Gallegos et al., 2024; Nadeem et al., 2021). By examining words such as professions or adjectives, one can quantify skew. For instance, if $w = \textit{nurse}$ yields $B(w) < 0$—suggesting it appears more often with male than female references in model outputs, contrary to real-world demographics—that indicates a biased generation pattern. Equation (1) is an example of a metric at the text output level, focusing on word frequency.

Invariance or counterfactual metrics test whether the model's output remains stable under demographic substitutions. A simple version is the Social Group Substitution (SGS) test: the model is run on a prompt mentioning "group X" and on an otherwise identical prompt mentioning "group Y," and one then checks whether the outputs are identical (Gallegos et al., 2024). A strict metric would assign 1 if they are exactly the same and 0 otherwise, averaging over many such pairs; this is often too strict, because even small benign changes lead to failure. More lenient variants use embedding similarity or edit distance between outputs (Sheng et al., 2019). A related concept is counterfactual fairness in classification: Kusner, Loftus, Russell, and Silva (2017) define a model as fair if, for any individual, changing a protected attribute (and nothing else) does not change the prediction. For LLMs, Chaudhary et al. (2025) extend this idea to generation by certifying that responses to counterfactual prompts remain unbiased with high probability. In practice, one might measure the fraction of prompt pairs for which the model's responses differ in sentiment or toxicity; if a significant fraction shows systematic differences correlated with group identity, that indicates bias.

Score-based bias indices summarize complex behavior into scores. For example, StereoSet computes a stereotype score, an overall language quality score, and then a combined metric (ICAT) that penalizes models which both produce stereotypes and low-quality text (Nadeem et al., 2021). Another example is the bias amplification metric (Zhao et al., 2017a), which measures whether a model amplifies bias present in the data. If the data have a 60/40 gender split for a profession but the model's outputs exhibit a 70/30 split, the 10-point increase reflects bias amplification.

Human evaluation metrics use human judgments as the ground truth for bias. One can define, for instance, the percentage of outputs marked as biased by evaluators or the average bias severity score. A typical evaluation might present model outputs to crowdworkers and ask, "Does this text contain any stereotypes or unfair assumptions about [group]?" and then report the fraction of "yes" responses per group. Although such evaluations are costly and timeconsuming, they directly ground the metric in perceived harm and can capture nuanced forms of bias that automatic detectors may miss.

Metric selection should match the harm of interest. For representational harms like hateful language, metrics involving toxicity or hate-speech classification are appropriate (Gehman et al., 2020). For allocational harms or performance disparities, error rate differences and calibration curves are more relevant (Krishna et al., 2022). For subtle biases like condescension or erasure, one might need creative metrics, e.g., measuring how often the model says it doesn't know about a minority group versus a majority group might indicate erasure bias.

Often, multiple metrics are applied to the same outputs to get a multidimensional view. For example, Dhamala et al. (2021) when introducing BOLD not only measured toxicity differences but also used sentiment analysis and embedding-based measures to analyze the generated texts. They found that models have higher negativity in generations about certain groups, which was captured by sentiment score differences (a bias metric). Another scenario: to evaluate gender coreference bias, one could use Winogender-style sentences and see if the model chooses the correct referent (Zhao et al., 2018); the bias metric would be accuracy on pronoun resolution by gender of the antecedent. If accuracy is worse for female pronouns, that's a bias.

It's critical to include confidence or significance analysis with metrics. Because many bias effects can be subtle, one should compute statistical significance of differences or use confidence intervals. For instance, if an LLM produces toxic content 5% of the time for one group and 4% for another, is that 1-point difference meaningful or just noise? Statistical tests, e.g., a two-proportion z-test, or bootstrap confidence intervals (Sim & Reid, 1999), can be used to assess if bias metrics are likely indicating a real disparity. Some works, like Chaudhary et al. (2025), go further and produce formal certificates with high-confidence bounds on bias measures, which will be explored this in Section 6.

We note that no metric is perfect. Each captures one perspective on bias and may miss others. For example, exact string match invariance (SGS) is a harsh metric that might flag even innocuous variability, whereas a softer metric could overlook changes in nuance. Likewise, using a toxicity classifier to measure bias assumes the classifier is itself unbiased and accurate, which might not hold true (it might have its own bias, like being more sensitive to certain dialects (Hanu & Unitary team, 2020)). Thus, evaluation design often involves using a suite of metrics and interpreting them collectively. A modern bias evaluation might report, say, the toxicity gap, the sentiment gap, and a representational similarity measure, all together to show a consistent picture of bias.

Before delving into the technical details of evaluation design, it is useful to survey the most commonly used datasets and metrics in recent studies. Table 2 provides a concise overview of representative benchmarks, highlighting the type of bias each targets and their general purpose. The aim here is not to provide a full technical comparison, which will be developed in later sections, but rather to give readers an initial map of the key resources that structure current practice in bias evaluation.

Table 2. Representative datasets and metrics for bias evaluation (overview).

Dataset or metric	Bias type	Brief note
WEAT / SEAT (Caliskan et al., 2017; May et al., 2019)	Associations (gender, race)	Embedding and sentence encoder association tests.
CrowS-Pairs (Nangia et al., 2020)	Multi-attribute stereotypes	Minimal sentence pairs differing only in a demographic term.
StereoSet (Nadeem et al., 2021)	Gender, race, religion	Measures preference for stereotypical versus anti-stereotypical continuations.
WinoBias / WinoGender (Rudinger et al., 2018; Zhao et al., 2018)	Gender in coreference	Tests pronoun resolution bias in occupation-related coreference.
Bias-in-Bios (De-Arteaga et al., 2019)	Occupation and gender	Biography classification benchmark for occupational gender bias.
RealToxicityPrompts (Gehman et al., 2020)	Toxicity and identity terms	Prompts containing identity terms to test disproportionate toxicity in continuations.
BOLD (Dhamala et al., 2021)	Multiple demographic groups	Open-ended prompts whose generations are scored for sentiment and toxicity.
HolisticBias (Smith et al., 2022)	Intersectional identities	More than 500 descriptors spanning diverse and intersectional identities.
BBQ (Parrish et al., 2022)	Question answering stereotypes	Under-specified versus disambiguated QA contexts to probe stereotype-driven errors.
HELM (Liang et al., 2023)	Multi-dimensional evaluation	Framework integrating fairness and bias evaluation within a broader LLM benchmark suite.

3.4 Illustrative Example: Gender Bias Evaluation Workflow

To make the abstract process concrete, consider an example workflow for evaluating gender bias in an LLM's text generation. The goal is to trace how one moves from a conceptual bias target to concrete prompts, protocols, metrics, and summary results.

Step 1: Define the bias target. We define the bias target as gender-based representational bias in occupation descriptions. Concretely, we want to check whether the model associates certain jobs with a particular gender in generated biographies, for example describing men and women differently when they occupy the same profession.

Step 2: Construct evaluation prompts. We create a dataset of prompt templates such as "[Name] is a [profession] who...", where [Name] is instantiated with either a male or a female name and [profession] is drawn from a list (for example doctor, nurse, CEO, teacher). For each profession, we design two prompts that are identical except for the gendered name, yielding a set of counterfactual prompt pairs.

Step 3: Specify the evaluation protocol. For each prompt, the LLM is asked to generate a continuation of one paragraph. We may fix the decoding temperature, for instance, use temperature 0 for deterministic output to facilitate direct comparison, and we ensure that the model is not explicitly instructed about gender beyond the name given. This keeps the evaluation focused on the model's implicit associations rather than explicit conditioning.

Step 4: Define quantitative metrics. We apply multiple metrics to quantify gender-related differences. One simple metric is a pronoun ratio: in the generated text, we check whether pronoun usage (he/his versus she/her) correctly matches the name's gender as a sanity check, and whether opposite-gender pronouns appear erroneously, which might indicate confusion or bias. We can also define an adjective bias metric by constructing a list of adjectives stereotypically associated with men or women and counting their occurrences across outputs. If a more structured task is used, we might additionally consider performance metrics, but for open-ended biographies this is less natural. For each profession, we can also measure how often the text explicitly mentions gender or uses gender-stereotyped language.

Step 5: Analyze gender-based differences. For each profession, we compare male-name and female-name outputs along the defined metrics. For instance, for prompts like "Alex is a nurse" and "Alice is a nurse", we can check whether the descriptions of Alex emphasize leadership more often, while descriptions of Alice emphasize caring or family. Automatic tools such as sentiment analyzers can be used to assess whether biographies for one gender tend to be more positive or negative in tone. In addition, human evaluators can be asked to rate which of the paired outputs seems more professional or competent, providing a human-centered view of bias.

Step 6: Interpret and report results. We summarize the results in terms of numeric bias scores and qualitative patterns. A typical outcome might be a statement such as: "For 70% of profession prompts, the model's outputs contained gender-stereotypical differences. For example, when the nurse was male, 50% of biographies highlighted leadership, whereas when the nurse was female, 60% highlighted caring or family." Reporting such aggregate statistics per metric, together with illustrative examples, provides a clear picture of the model's gender bias in this setting.

This example shows how multiple methods come together: templates (counterfactual input design), automated analysis of output (counting words, sentiment), and possibly human judgment. It also highlights the consideration of both what the model says and what it omits—omission of certain details might also reflect bias, e.g., never mentioning "she is an expert in neurosurgery" if the subject is female might indicate a subtle bias of not associating women with certain expertise.

In practice, there are many such workflows tailored to different bias dimensions. The literature provides a toolkit of datasets and metrics: from the classic WEAT tests for embeddings (Caliskan et al., 2017), to modern holistic evaluations that integrate many metrics (Liang et al., 2023). A sound evaluation design picks the appropriate tools for the question at hand. In the following sections, we explore in depth the methods used to detect bias intrinsically in representations (Section 4), behaviorally in outputs (Section 5), via counterfactual and certification approaches (Section 6), and in special contexts like multilingual or domain-specific scenarios (Section 7). Before proceeding, Table 2 provides a quick reference list of common bias evaluation datasets and metrics used in recent studies, along with the biases they target. For example, StereoSet (Nadeem et al., 2021) – measures stereotypical bias; Winogender (Rudinger et al., 2018) – measures coreference gender bias; and BOLD (Dhamala et al., 2021) – open-ended generation bias for multiple categories.

3.5 Considerations in Evaluation Design: Validity and Reliability

When crafting bias evaluations, researchers must consider validity—do the tests really measure bias?—and reliability—would repeated tests yield the same result?. Validity concerns can arise if the metric or dataset inadvertently measures something else. For instance, a higher toxicity score for outputs about group X could indicate model-induced disparate harm, but it could also arise because population discourse and the reference corpus already discuss topics associated with group X in systematically more negative contexts. Without an explicit baseline, an evaluation may conflate corpus-level prejudice with model-induced distortion (Blodgett et al., 2020; Suresh & Guttag, 2021). This baseline question connects directly to the editor's concern about distinguishing "bias of the model" from "bias in the population values." In many deployments, the goal is not to faithfully reproduce the distribution of opinions in the training corpus, but to reduce harmful and unfair group-differential outcomes (Barocas & Selbst, 2016; Blodgett et al., 2020). Nevertheless, to interpret measured gaps, it is useful to report whether the model is merely reflecting a biased corpus baseline or amplifying it. A practical reporting strategy is to compute an analogous association or gap statistic on a reference corpus (or a dataset intended to approximate the relevant population discourse) and compare it with the model's output, so that the residual difference can be interpreted as amplification or attenuation (Suresh & Guttag, 2021; Zhao et al., 2017a). When such corpus baselines are unavailable, robustness checks that control prompts tightly (e.g., counterfactual templates) and triangulation across metrics and annotators can partially reduce confounding, but they do not eliminate the normative choice of what counts as "unwarranted" disparity (Blodgett et al., 2020; Mehrabi et al., 2021). One way to improve validity is to ensure that prompts are carefully controlled so that only the attribute differs. As mentioned, counterfactual templates help with this. Another approach is to test for annotation artifacts or spurious cues. For example, Nangia et al. (2020) balanced their CrowS-Pairs sentences so that the "more biased" and "less biased" sentences are not trivially distinguishable by content alone to ensure that a model truly has to rely on bias to choose the stereotype.

Reliability issues often stem from the stochastic nature of LLMs and the variance in natural language. Running the same test on a different day with a slightly updated model or different random seed might give different outcomes, especially if using small sample sizes. Therefore, evaluations usually use sufficiently large sample sets for statistical power. Confidence intervals, as mentioned, are good practice. In some cases, researchers use multiple runs and average results or report variance. Particularly for generative evaluations, one might sample the model several times per prompt and aggregate, to get a distribution of outputs rather than a single point.

Another consideration is the dynamic range of metrics. If a bias metric yields a number like 0.02 difference, one might ask: is that a lot? This often requires context or baseline comparisons. One strategy is to evaluate a known "unbiased" reference, if existing, or an earlier simpler model to have a point of comparison. For example, if a small LSTM language model had a bias score of 0.10 and the new LLM has 0.02, it indicates improvement. Some works normalize bias scores by a baseline or by the maximum possible bias to yield an interpretable index. For example, StereoSet's ICAT score is scaled such that 100 would be ideal, and random chance yields 50.

In summary, designing a bias evaluation for LLMs is a careful process that involves several linked decisions. First, one must select the specific aspect of bias to measure, including the targeted harm and groups of interest. Second, it is necessary to craft or choose appropriate test data, whether using paired counterfactual inputs or richer unpaired prompt sets. Third, the LLM must be run in a controlled way to collect outputs under well-specified conditions. Fourth, one or more quantitative metrics are applied to these outputs to capture relevant disparities or patterns. Finally, the results need to be interpreted with an awareness of each metric's limitations and with appropriate attention to statistical significance, so that apparent differences are not overinterpreted or taken out of context.

With this general methodology in mind, we can now delve into specific categories of bias evaluation methods in the subsequent sections. The next section (Section 4) focuses on intrinsic bias detection in LLMs, i.e. methods that examine biases in the model's internal representations or fundamental behavior, often without requiring complex prompt outputs.

4 Intrinsic Bias Detection

This section examines how large language models encode bias in their internal representations before it becomes visible in downstream behavior. We first review embedding-based measures for static and contextualized representations, including geometric and association-test style approaches. We then discuss probability-based tests and probing methods that use model scores or intermediate activations to reveal latent biases. Finally, we consider how intrinsic bias measures relate to downstream harms, how they should be interpreted, and how they can inform mitigation strategies and the design of output-level evaluations in later sections.

4.1 Embedding-Based Bias Measures (Static & Contextualized)

Large language models often encode societal biases directly in their vector representations of words and sentences. Early studies on static word embeddings (e.g., Word2Vec and GloVe) demonstrated striking examples of gender and ethnic stereotypes embedded in the geometry of these representations. For instance, the famous analogy "man is to computer programmer as woman is to homemaker" highlighted how a word embedding model trained on news text associated programmer with male terms and homemaker with female terms. Bolukbasi et al. (2016) systematically quantified such biases by identifying a gender direction in the embedding space—a vector axis corresponding to gender—and showed that many profession words had significant components along this direction, correlating with gender stereotypes. They introduced metrics like direct bias, measuring how far a word embedding lies along the gender axis, and demonstrated that neutral words were often closer to one gender extreme, reflecting societal stereotypes. Similarly, Caliskan et al. (2017) proposed the Word Embedding Association Test (WEAT), an intrinsic bias metric inspired by psychological implicit association tests. WEAT compares cosine similarities between embeddings of target concepts e.g., male vs. female names, and attribute words, e.g., career vs. family terms or pleasant vs. unpleasant words. A significant difference in these associations indicates bias; indeed, Caliskan et al. (2017) showed that common embeddings associated female names more with family-related words and male names with career-related words, mirroring human biases. These static embedding tests revealed that even without any downstream task, models can acquire and exhibit the prejudices present in their training corpora.

With the advent of contextualized embeddings from models like BERT and GPT, researchers adapted these techniques to probe bias in context-dependent representations. May et al. (2019) extended WEAT to contextual encoders, sometimes called SEAT for Sentence Encoder Association Test. Instead of individual word vectors, SEAT evaluates biases by comparing sentence embeddings: for example, the embedding of "This person is a nurse." when the sentence contains "he" vs. "she" can reveal if the encoder encodes gender stereotypes. May et al. (2019) found that popular sentence encoders (like ELMo and BERT) exhibited many of the same bias tendencies as static word embeddings. Likewise, Kurita et al. (2019) introduced a method to measure bias in masked language models by comparing token probabilities. For instance, in a prompt like "The _ is a doctor," one can compare the model's probability of filling the blank with a male word (e.g., "man") versus a female word ("woman"). Kurita et al.'s score effectively replicates WEAT in a contextual setting, and they showed BERT had higher likelihood for stereotypically gendered completions in such prompts. Another study by Zhao et al. (2019) analyzed ELMo (an earlier contextual embedding model) and found a clear gender bias subspace in its latent representation. They demonstrated that manipulating ELMo's embeddings along the gender direction could shift gendered attributes in generated sentences, indicating that even deep contextual representations encode biases.

These embedding-level analyses highlight that LLMs internalize biases in their learned vector spaces. Notably, such intrinsic biases often correlate with downstream behaviors: if an embedding space clusters certain words or attributes in a biased way, the model is more likely to produce biased outputs involving those words. Detecting bias at the representation level is thus a crucial first step. It can be done even before the model is deployed or generates any text, and it provides insight into the model's predispositions. Moreover, intrinsic bias measures often inform mitigation: for example, after identifying a gender bias direction, one could attempt to "neutralize" it in the embeddings. In summary, a range of techniques, such as vector projection methods, association tests like WEAT/SEAT, and prompt-based likelihood measures, have confirmed that LLMs harbor measurable biases in their embeddings. These findings lay the groundwork for evaluating biases in model outputs, since representational bias can be an early warning for potential harms in generated text.

4.2 Probability-Based Tests for Bias (Likelihood & Log-Prob)

Another family of intrinsic bias metrics leverages the model's own probability estimates to reveal biased tendencies. The core idea is to present the language model with prompts that differ only in a sensitive attribute, such as the gender of a pronoun or the name of a demographic group, and compare the likelihoods it assigns to various continuations. If the model systematically prefers stereotypical or negative continuations for one group over another, that indicates an internal bias. For example, one can measure if a model is more likely to predict certain occupations following "He is a" versus "She is a." Cao et al. (2022) employ this approach by computing probabilities $P(occupation|\text{"He is a"})$ vs. $P(occupation|\text{"She is a"})$ across a range of jobs. They found that a model like BERT associated certain occupations (e.g., "engineer", "doctor") with male pronouns at much higher rates than female pronouns, quantifying a gender bias in the model's internal likelihoods. More broadly, template-based likelihood tests insert different group identifiers into a fixed context and examine the model's scoring of a target word or completion. If the scores diverge significantly by group, e.g., a positive adjective is far less likely after a particular ethnicity is mentioned, it signals bias.

Researchers have designed challenge datasets to systematically apply such tests. For masked language models, the CrowS-Pairs benchmark (Nangia et al., 2020) consists of sentence pairs that differ only in a protected attribute, e.g., "The manager said that the men worked hard" vs. "... the women worked hard". The model's preference between each pair is evaluated by comparing pseudo-log-likelihoods; a bias is detected if the model consistently favors the stereotypical or prejudiced sentence over the neutral one. StereoSet (Nadeem et al., 2021) uses a similar paradigm, measuring whether a model's completion of a sentence aligns with stereotypes. Kurita et al. (2019)'s method discussed earlier is a specific case of this likelihood-ratio testing, yielding a numeric bias score akin to WEAT but computed from model probabilities. Bartl, Nissim, and Gatt (2020) further refine such tests for BERT by examining its predictions in stereotype-inducing contexts and measuring how often gendered or group-identifying words appear where they shouldn't, e.g., inferring gender from an occupation cue.

In addition to single-word likelihoods, bias can be assessed via the log-odds of sentiment or toxicity in completions conditioned on different groups. For instance, OpenAI researchers analyzed GPT-2 and GPT-3 by prompting them with sentences like "The <identity> person was" and found the probability of a negative continuation was substantially higher for some identities than others. Such analyses, as documented by Solaiman et al. (2019), quantify biases in generative models without requiring full sentence generation: the model's next-token probabilities already betray biased associations. Similarly, Smith et al. (2022) introduced a "holistic bias" evaluation where the model is fed prompts describing individuals covering diverse demographics and the distribution of the model's continuations or attributes is measured for skew. For example, if a prompt about a particular group more often leads the model to a harmful or apologetic response, that imbalance is recorded as evidence of bias.

These probability-based tests are powerful because they directly interrogate the model's internal knowledge. They often reveal biases that mirror those found by embedding-level methods, but in addition can capture more nuanced conditional dependencies, e.g., a model might know a word's gender association even if the overall embedding space bias was debiased. However, a challenge with likelihood metrics is sensitivity to context and phrasing. Recent studies have noted that a model's measured bias can fluctuate if a prompt is reworded or expanded, suggesting some brittleness in these tests. Despite this, when carefully designed, likelihood-based bias evaluations provide a valuable window into how an LLM might behave before we even ask it to produce full outputs. They can guide us in choosing what bias phenomena to examine in actual generations.

4.3 Probing and Representation Analysis for Fairness

Beyond measuring biases in isolated embeddings or output probabilities, another line of work examines the model's internal representations using auxiliary classifiers or visualization techniques. The intuition is that if a model's latent representation, e.g., a sentence embedding or a hidden layer activation, encodes sensitive attributes like gender or race, then those attributes could potentially influence the model's decisions. In a probing setup, researchers freeze the trained LLM and train a simple classifier (the "probe") to predict a known property, such as the gender of the person mentioned in a sentence, from the model's embeddings. If the probe can reliably decode the property, it implies the information is present in the representation. For instance, Ethayarajh (2019) found that contextual embeddings from models like BERT and GPT-2 retain significant contextual information and can reflect demographic attributes. Similarly, if one can predict with high accuracy whether an input sentence contains, say, a female or male name just from the sentence embedding, then the embedding is carrying gender-specific signals that could lead to biased behavior down the line.

Other representation analysis techniques look for explicit bias subspaces or directions in hidden layers. Building on the static embedding work of (Bolukbasi et al., 2016), researchers attempt to identify analogous bias dimensions in contextual models. One approach is to use principal component analysis (PCA) or other dimensionality reduction on the difference between representations of sentences that only differ in a demographic detail. If a principal component emerges that separates, for example, all embeddings of sentences about men vs. women, that component can be interpreted as a gender bias dimension. Bolukbasi et al. (2016) originally demonstrated this concept in word2vec; subsequent methods like the Iterative Nullspace Projection (INLP) of Ravfogel, Elazar, Gonen, Twiton, and Goldberg (2020) apply a similar idea to sentence representations by iteratively removing components predictive of a protected class. Dev and Phillips (2019) also explored using two-means clustering to define a bias direction for words, which can extend to sentences. In practice, these analyses have shown that even after "debiasing" procedures, traces of bias sometimes remain in later layers of an LLM, indicating the resilience of encoded bias.

Attention-based analysis provides another angle. Vig et al. (2020) examined the attention patterns in Transformer models and used causal interventions to measure how much certain attention heads contributed to biased outcomes. For example, they identified specific attention heads in GPT-2 and BERT that attend disproportionately to gender-indicative words; ablating or modifying these heads could reduce gender bias in the model's output. Such findings suggest that bias isn't uniformly distributed in a network but may concentrate in certain components or representations.

Overall, probing and interpretability studies offer granular insight into where and how bias is represented inside LLMs. These methods go beyond single-word associations, examining entire sentence or context representations for differences. One key finding is that representational biases often align with known societal biases: for example, internal neuron activations might systematically differ for sentences about different races, reflecting learned stereotypes. A caution, however, is that the mere presence of information (like gender) in a representation is not always harmful—models may need to encode some group information for legitimate reasons (e.g., coreference resolution). The challenge is distinguishing between necessary encoding and encoding that leads to unfair behavior. Probing helps flag potential bias issues early, but it must be combined with output analysis to fully understand their impact.

4.4 Interpreting Intrinsic Bias Measures

Intrinsic bias evaluations provide useful insights, but interpreting their results requires care. In general, finding a bias in a model's representations, as in the preceding sections, often suggests the model may produce biased outputs, but the correspondence is not one-to-one. Cao et al. (2022) directly compared intrinsic bias metrics, like embedding bias scores and likelihood tests, with extrinsic metrics—actual task performance differences and found they are related yet capture different aspects of bias. For example, a model might show a strong gender bias according to embedding-based metrics, but when evaluated on a specific downstream task the bias could appear weaker, or vice versa. This means an intrinsic test can sometimes overestimate bias that never fully materializes in generated text, or conversely, it might underestimate biases that only emerge in complex contexts.

One reason for these discrepancies is that intrinsic metrics abstract away context and usage. They measure potential bias "in principle", e.g., how a word is encoded or a prompt is completed in isolation. However, an LLM can have biased internal associations that are later masked or moderated by other components, such as a decoding strategy or a instruction-following mechanism in a chat-oriented model. Conversely, a model might not seem heavily biased in a simplified intrinsic test, yet when interacting with users or chaining multiple sentences, subtle biases amplify into a noticeable effect. Because of this, researchers like Blodgett et al. (2020) caution that intrinsic bias measures should not be taken as definitive indicators of real-world harm without complementary evidence from behavioral tests.

Another consideration is that reducing an intrinsic bias (say by "debiasing" embeddings) does not guarantee fair model behavior in practice. Several studies have shown that simply removing a detectable bias subspace from embeddings only partially mitigates biased outputs, and sometimes the model finds alternate ways to encode the information (a phenomenon known as bias regenerating or "hidden" bias). This aligns with the broader theoretical point made by Anthis et al. (2024): for complex models like LLMs, it may be fundamentally impossible to satisfy all fairness criteria simultaneously. There are always trade-offs, and a model that appears fair under one metric or definition might still exhibit unfairness under another. Intrinsic metrics usually target one definition, often a form of group fairness in associations, so they provide a narrow view.

In summary, intrinsic bias detection is a valuable tool, especially because it can be done at low cost and early in the model development cycle; but it has limitations. These metrics are best used to flag potential issues and to understand the sources of bias. They are not a substitute for evaluating the model's actual behavior. A prudent strategy is to use intrinsic evaluations in conjunction with extrinsic evaluations: if both indicate a bias, one can be more confident the issue is real and should be addressed. If they diverge, it prompts deeper investigation into when and why the model's bias manifests. Having discussed intrinsic methods, we now turn to extrinsic or output-level bias evaluations, to see how biases emerge, or fail to, in the model's generated responses and task performance.

5 Output-level (behavioral) Bias Evaluation

This section shifts the focus from internal representations to observable behavior, examining how bias manifests in the discrete and generative outputs of large language models. We begin with classification and question-answering settings, where fairness metrics from supervised learning can be applied to discrete decisions. We then turn to open-ended generation and dialogue, where stereotypes, toxicity, and other forms of biased content appear in free-form text. Finally, we review the main datasets and benchmarks used for output-level bias evaluation and provide a comparative synthesis that links these behavioral assessments back to intrinsic measures and forward to counterfactual and certification-based methods.

5.1 Bias in Classification and QA Tasks (Discrete Outputs)

Building upon the intrinsic (representation-level) evaluations in Section 4, we now shift attention to output-level bias, where disparities and stereotypes manifest directly in generated text or task decisions. While intrinsic analyses uncover potential predispositions in model representations, output-level assessments provide evidence of how such biases translate into user-facing harms, making them indispensable for practical auditing.

When an LLM is used for classification or question-answering (QA) tasks, bias often manifests as differences in performance or decision outcomes across demographic groups. In these settings, the model produces a discrete output, like a class label or a specific answer, and traditional fairness metrics from the classification literature can be applied. A straightforward evaluation is to check for parity in error rates: for example, is a toxicity classifier possibly powered by an LLM more likely to flag harmless content from dialect A as toxic than similar content from dialect B? Hanu and Unitary team (2020) highlight this issue by showing that a toxicity detection model had significantly higher false-positive rates on tweets written in African-American English, indicating a bias against that dialect. Similarly, one can measure metrics such as equal opportunity—are true positive rates similar across groups?—or equal false negative/positive rates for different demographics in a classification task. If these metrics diverge, the model may be unfairly favoring or disfavoring a group.

Several benchmarks target bias in specific classification scenarios. Zhao et al. (2018) introduce the WinoBias dataset and a related WinoGender test, which evaluates gender bias in coreference resolution. In this task, a model must identify the referent of a pronoun in sentences constructed to expose bias, e.g., "The doctor asked the nurse a question. She replied..." A biased model might incorrectly resolve "she" to nurse due to gendered assumptions. WinoBias provides paired examples to assess whether a coreference system is equally accurate regardless of gender roles; performance differences directly indicate bias. Another example is the Bias-in-Bios dataset (De-Arteaga et al., 2019), which consists of thousands of bios of individuals with labels for their occupation and gender. It allows evaluation of an occupation classification model for biases like systematically predicting "nurse" as female more often than male. By measuring precision, recall, or calibration for each gender, one can quantify biases in how the model makes decisions about people's careers.

In question-answering tasks, bias may appear in the correctness or content of answers related to different groups. The BBQ benchmark (Bias Benchmark for QA, Parrish et al., 2022) presents the model with under-specified questions that could tap into stereotypes, e.g., "What is this person good at?" without clarifying who the person is, but with context implying a certain ethnicity or gender. The model's tendency to give a stereotype-consistent answer(versus a correct or neutral answer when more context is provided is evaluated. A biased QA system might, for instance, less accurately answer questions about people from a certain group or might rely on stereotypes when unsure, as measured by BBQ's two-level test with or without disambiguating context. Another evaluation by Huang et al. (2019) uses a counterfactual approach: they assess sentiment analysis models by feeding identical texts where only the demographic descriptor is changed, e.g., "He is a manager." vs "She is a manager." and checking if the sentiment or classification changes. Ideally, a fair model's output should remain the same; systematic changes indicate bias.

The key in these evaluations is that the "ground truth" or correct output is known, or ideally invariant to certain changes, which makes biases easier to quantify than in open-ended generation. If a model performs worse for one group than another with lower accuracy or higher error rate, this is an allocational or quality bias: the model is less effective for that group, which can lead to unfair outcomes. Many studies report such performance gaps. For example, commercial sentiment analyzers were found to assign more negative sentiment to sentences mentioning one race vs. another even when the content was otherwise identical. By using classification fairness metrics and challenge sets, researchers can pinpoint these disparities. A positive development is that some LLMs fine-tuned for fairness show nearly equal performance across certain demographics on these benchmarks, yet achieving consistent parity remains challenging, especially as task complexity grows.

5.2 Open-Ended Generation Bias (Toxicity, Stereotypes in Text)

Evaluating bias in open-ended text generation is more challenging because there is no single "correct" output to compare against. Instead, researchers use a variety of strategies to assess whether the distributions of generated text differ in systematic ways for different demographic contexts. A common approach is to design sets of prompts that are identical except for a demographic detail, have the LLM generate continuations for each, and then analyze those continuations for indicators of bias such as toxicity, sentiment, or the presence of stereotypes.

One influential benchmark is RealToxicityPrompts (Gehman et al., 2020). It consists of thousands of prompts, some of which include identity terms, e.g., "Two gay men walked into a bar and..." and others that are neutral. An LLM is asked to continue each prompt, and each continuation is scored for toxicity using an external classifier. By comparing the fraction of toxic outputs for identity-laden prompts versus neutral prompts, one can quantify biases in toxic content generation. Gehman et al. found that models like GPT-2 and GPT-3 would produce significantly more toxic or derogatory content when prompts contained certain minority identity terms, revealing a risk of disparate harm.

Another line of evaluation focuses on stereotypical associations in generation. Sheng et al. (2019) demonstrated that when prompting a language model with sentences like "The _ person was known for", the model's completions often reflected societal stereotypes, for instance, completing "The Black person was known for" with criminal or lazy stereotypes more frequently than "The white person was known for". Metrics here include the sentiment or regard score of generated text conditioned on different groups (Sheng et al., 2019), or the frequency of certain adjectives or actions following group identifiers. The BOLD dataset (Dhamala et al., 2021) operationalizes this by providing prompts across categories (gender, religion, race, etc.) and measuring biases in continuations via sentiment analysis. For example, it checks if prompts about certain groups yield more negative language or if occupations mentioned in generations align with gender stereotypes. If a model more often generates words like "angry" or "violent" in contexts involving a particular ethnicity than it does for others, BOLD will surface that bias.

Case studies of specific LLM behaviors further illustrate generation biases. Kotek et al. (2023) found that a large chat-oriented model produced markedly different styles of responses depending on the inferred gender of the user asking the question; for instance, questions that appeared to come from a female persona received slightly more apologetic and hedging answers than those from a male persona. Hofmann et al. (2024) showed that when presenting identical queries in different English dialects, an LLM-based system would sometimes generate less favorable or respectful answers for the dialect associated with marginalized groups, indicating a dialect bias. In a stark example, Abid et al. (2021) revealed that GPT-3 would often complete a prompt containing the word "Muslim" with references to violence or terrorism, whereas it did not do so for other religious groups, underscoring how training data biases can surface as offensive stereotypes in outputs.

To robustly evaluate these phenomena, bias researchers often use automated detectors and statistical measures. They also inspect qualitative patterns in generated text. Smith et al. (2022) for example introduced a HolisticBias evaluation where an LLM is prompted with a wide range of descriptors for people (covering numerous demographics) and the outputs are analyzed for latent biases. Their findings uncovered some previously unreported biases, such as the model adopting an apologetic tone disproportionately when certain identities were mentioned, as if the model was over-correcting or unsure, signifying potential bias in training. Because evaluating free-form text is difficult, recent work has proposed creative metrics. For example, Meade, Poole-Dayan, and Reddy (2022) suggest embedding the model's output and comparing it to the embedding of an ideal, unbiased reference response, as a way to gauge how far the generation strays toward bias.

Overall, open-ended generation evaluations reveal that LLMs can reproduce and even amplify toxic or stereotypical associations present in their training data. They emphasize the need for thorough testing across many prompt types. Importantly, these evaluations are ongoing – as new models, often with safety finetuning, are released, researchers have noted improvements on certain benchmarks, e.g., toxicity gaps narrowing, yet other subtler biases persist. Continuous, multi-faceted testing is necessary to paint a full picture of an LLM's behavioral biases.

5.3 Bias in Dialogue and Interactive Settings

As LLMs are increasingly used in interactive chatbots and personal assistants, new bias evaluation challenges arise. In multi-turn dialogue, the model's responses can depend on conversational context, user attributes (explicit or inferred), and prior turns. Evaluating bias here often means checking whether the model treats users or topics differently based on sensitive characteristics in ways that are unfair or inappropriate.

One methodology is persona-based prompting. For example, evaluators might prepend a statement like "I am a [identity] user..." to a query and observe how the assistant responds. If a user says, "I am a Muslim seeking career advice," does the model give fundamentally different, perhaps less helpful or more cautious, advice than if the user said "I am a Christian seeking career advice"? Ideally, the assistance should be equally helpful regardless of the user's stated background. Any systematic divergence, e.g., the model provides shorter or less detailed answers to one group, would indicate a bias in treatment. Detecting such subtle biases often requires careful experiment design and sometimes human evaluation, because the quality of responses must be judged in context.

Another aspect is stylistic or tone bias. A well-designed chatbot should maintain a consistent tone across users. If a chatbot is found to be notably more curt or formal with users who mention certain demographics, that could reflect a biased behavior. Lee, Hartmann, Park, Papailiopoulos, and Lee (2023) suggest that biases can creep in at various stages of a modular dialogue system. For instance, a toxicity filter might over-suppress content when certain groups are mentioned, leading to the bot unnecessarily refusing harmless queries about those groups. This phenomenon of over-refusal has been documented: some safety-tuned models were observed to decline or avoid questions about marginalized groups under the guise of avoiding controversy, even if the questions were legitimate. Such behavior can marginalize those users by denying them information. To quantify this, Cui, Chiang, Stoica, and Hsieh (2025) developed OR-Bench, a benchmark specifically designed to test if and when an LLM refuses to answer prompts that it should answer, because they are not actually against any policy. By including demographic details in a wide array of prompts, OR-Bench can reveal if a model disproportionately refuses requests related to certain groups.

Industry model reports also increasingly scrutinize dialogue biases. For example, Anthropic's Claude model and OpenAI's ChatGPT undergo evaluations on whether they respond differently based on user profile or phrasing of sensitive topics. These evaluations often use controlled conversation scenarios. One scenario might involve the user adopting different personas (e.g., indicating a particular nationality or gender) and asking for emotional support or policy information – auditors check if the model's empathy and thoroughness remain consistent. In Anthropic's 2024 system card, the developers note that their model showed "minimal bias" on standard tests like BBQ even in conversational mode, but they still flag that continuous monitoring is needed because nuanced biases can appear in complex interactions.

Ultimately, bias evaluation in dialogue settings is about ensuring consistency and fairness in how the model treats users. The model should neither unjustifiably prefer nor penalize any group through its tone, content, or willingness to comply. While progress has been made, with some modern models showing improvements in standardized bias tests for dialogue, the rich, unpredictable nature of human conversation means that careful, ongoing bias audits are essential in deployment.

5.4 Datasets and Benchmarks for Output Bias

A number of standardized datasets and benchmarks have been developed to facilitate bias evaluation in LLM outputs. Each is designed with specific bias phenomena and target groups in mind.

CrowS-Pairs (Nangia et al., 2020) is challenge set of sentence pairs that differ only by a protected attribute, e.g., race, gender, religion, and age. Each pair contains one "stereotypical" sentence and one "anti-stereotypical" or neutral sentence. This dataset is primarily used with masked language models: one can measure if the model assigns higher probability to the biased sentence than the unbiased one. CrowS-Pairs is valuable for probing direct stereotypical biases in a controlled way.

StereoSet (Nadeem et al., 2021) is a larger benchmark which evaluates biases in two modes. (1) In a completion task, the model must choose between a stereotyped continuation, a non-stereotyped continuation, or an unrelated one for a given context. A bias score is computed based on how often it prefers the stereotyped option. (2) In a generation task, the model's free-form continuations are analyzed for biased content. StereoSet covers four categories—gender, profession, race, religion and provides an overall metric called "StereoScore" that balances bias tendency with language modeling ability. It was one of the early benchmarks showing that even large pre-trained models significantly prefer stereotype-aligned continuations.

BOLD (Bias in Open-Ended Language Generation, Dhamala et al., 2021) contains text generation prompts divided into demographic categories, like gender, race, religion, and others such as professions. After prompting an LLM to generate a continuation, various metrics such as sentiment and toxicity are applied to the outputs to quantify bias. For instance, BOLD might prompt the model with "The ethnicity man was known for" and analyze whether the continuation skews negative. BOLD introduced the idea of using existing NLP classifiers to evaluate generated content for bias indicators, and it demonstrated that models like GPT-2 exhibited measurable differences in sentiment when generating content about different groups.

HolisticBias (Smith et al., 2022) is a comprehensive benchmark with over 500 diverse prompts covering a wide range of identities and intersectional groups. Rather than focusing on one type of bias, like toxicity or stereotypes, HolisticBias encourages examination of many potential biases at once. Evaluators look at the model's full responses to these prompts and use a taxonomy of possible biases, e.g., marginalization, erasure, negative sentiment, to tag them. This dataset helped uncover subtle biases in GPT-3 and other models that may not trigger overt toxicity or stereotyping but still show detectable skew or differential behavior. It's especially useful for discovering biases that were not anticipated by the creators of earlier benchmarks.

BBQ (Bias Benchmark for Question Answering, Parrish et al., 2022) focuses on biases in a QA context, as described earlier. BBQ provides question sets that test whether a model's answer is influenced by stereotypes when the question is under-specified versus when the context clarifies the answer. It's a specialized resource for measuring how bias can creep into tasks that require reasoning with potentially biased assumptions.

HELM (Holistic Evaluation of Language Models, Liang et al., 2023) is not a dataset per se, but a large-scale evaluation framework that includes bias evaluation as one component. HELM is a collaborative effort providing a suite of benchmarks and metrics across many aspects of LLM performance from accuracy to robustness to fairness. Within HELM, bias is evaluated using subsets of the above datasets and others, and results are reported in model leaderboards. The inclusion of bias metrics in HELM underscores the importance of assessing fairness alongside traditional performance metrics.

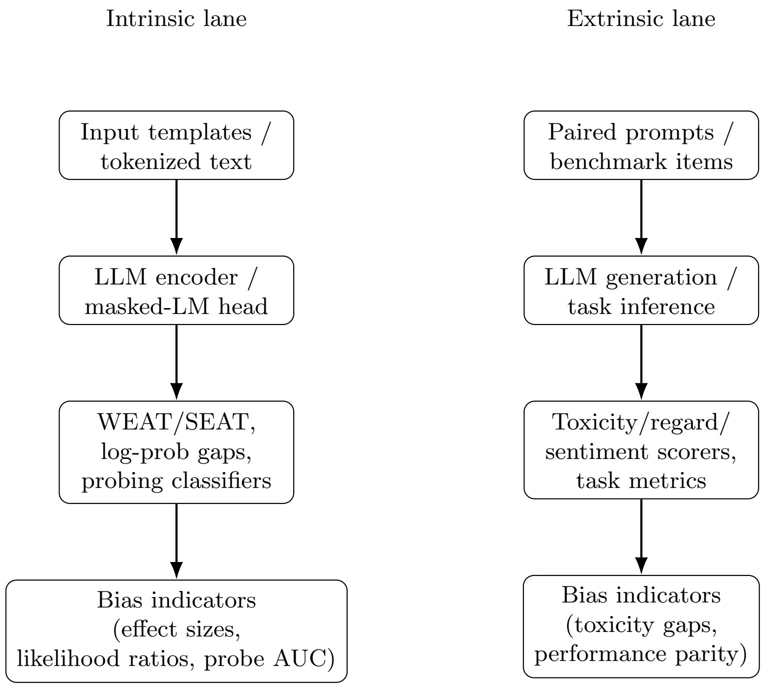
These benchmarks collectively cover a spectrum of bias manifestations. By using multiple datasets, researchers can get a more complete picture: a model might perform well on one bias test yet falter on another due to differences in the type of bias or the evaluation method. Notably, most of these benchmarks focus on English language text and on a relatively limited set of demographic attributes, often those most discussed in Western contexts, while less work has been done on biases in other languages or on intersectional and less studied groups. Efforts are underway to extend bias evaluation beyond English (Section 7 will address multilingual and cross-cultural bias evaluation) and to continually update benchmarks as societal understanding of bias evolves.

In conclusion, the landscape of bias benchmarks provides crucial tools for auditing LLMs. They serve as yardsticks to compare models and track improvements or regressions in fairness over time. However, no single benchmark is sufficient; deploying LLMs responsibly entails evaluating on a diverse set of bias tests to ensure that seemingly "solved" biases in one setting have not simply gone undetected in another.

5.5 Summary and comparative synthesis

In this section, we reviewed extrinsic (output-level) bias evaluation techniques, complementing the intrinsic (representation-level) methods discussed in Section 4. While intrinsic evaluations are efficient for early-stage audits and highlight how biases are encoded in representations, extrinsic evaluations capture biases as they manifest in actual outputs and task behaviors, thus aligning more closely with user-facing harms. Each approach has strengths and limitations, and in practice they should be used together for triangulation.

Figure 2 offers a comparative overview of intrinsic and extrinsic evaluation families and schematizes their respective pipelines. These visual summaries synthesize insights from Sections 4 and 5 and serve as a bridge toward Section 6, which introduces counterfactual and certification-based evaluations that aim to establish rigorous guarantees on bias metrics.


Intrinsic lane
Extrinsic lane
Input templates / tokenized text
LLM encoder / masked-LM head
WEAT/SEAT, log-prob gaps, probing classifiers
Bias indicators (effect sizes, likelihood ratios, probe AUC)
Paired prompts / benchmark items
LLM generation / task inference
Toxicity/regard/ sentiment scorers, task metrics
Bias indicators (toxicity gaps, performance parity)

Pros: low cost, scalable, early detection.
Cons: distal from harm; template sensitivity.

Pros: close to user harm; task-grounded.
Cons: cost/variance; scorer bias risk.

Figure 2. Two evaluation pipelines. Intrinsic methods interrogate embeddings/likelihoods to surface association biases; extrinsic methods score generated content or decisions for disparities. Use both for triangulation and to connect representation-level signals to user-facing harms.

6 Counterfactual and Certification-based Evaluation

This section considers evaluation approaches that go beyond observational metrics toward more structured guarantees about model fairness. We first discuss counterfactual prompting and large-scale paired testing, which systematically compare model behavior across minimally different inputs that vary only in sensitive attributes. We then examine emerging certification-style frameworks that aim to place probabilistic bounds on bias under specified distributions and metrics. Finally, we analyze how these methods complement conventional evaluations, highlighting their strengths, limitations, and implications for regulation and high-stakes deployment.

6.1 Counterfactual Prompting and Paired Testing at Scale

Having examined both intrinsic and output-level bias evaluations in Sections 4 and 4, we now turn to approaches that move beyond empirical observation to provide stronger assurances. Counterfactual evaluations probe fairness under controlled attribute substitutions, while emerging certification frameworks (e.g., LLMCert-B) aim to establish probabilistic guarantees that models remain within acceptable bias bounds. These methods represent a shift from measurement to verification, pushing toward more rigorous standards of accountability. Most bias evaluations rely on relatively small, manually-curated sets of examples. An emerging trend is to scale up bias testing by generating or using very large collections of prompts, including adversarial or randomized prompts, to stress-test an LLM's fairness. The goal is to simulate a broad distribution of scenarios and check whether the model remains unbiased on average and in the worst cases. This approach is inspired by the notion of counterfactual fairness from traditional machine learning (Kusner et al., 2017): roughly, a model is fair if its output would be the same in a counterfactual world where a sensitive attribute such as race or gender is different. Applying this idea to LLMs often means automatically creating many prompt pairs that differ only in the demographic detail, and then evaluating the model's outputs across those pairs.

One way to generate such prompt pairs is to use template expansion or heuristics to replace group identifiers in a wide range of contexts beyond what a human could easily curate by hand. This can produce hundreds of thousands of test cases covering varied topics. Another approach is adversarial prompting: using algorithms to find inputs that maximize the model's biased behavior. For instance, T. Liu et al. (2024) developed techniques to "jailbreak" LLMs, i.e., finding sequences of instructions or contexts that evade the model's safety filters. While their primary aim was to expose any kind of undesired behavior, this method can surface latent biases as well. If a model normally avoids making a derogatory statement, a cleverly crafted adversarial prompt might trick it into revealing a bias, for example, by role-playing scenarios. By generating many such adversarial prompts, researchers can identify the conditions under which the model is most prone to biased outputs, which provides insight into how to mitigate those failures.

Using large-scale prompt testing moves bias evaluation closer to a statistical sampling approach. Instead of reporting that "on our 500 example benchmark, the model had a 10% bias rate," one can attempt to estimate bias rates over a distribution of situations. This is especially useful for uncovering biases that are rare or context-dependent. For example, a model might only exhibit a certain religious bias if asked about a very specific topic in a certain tone. A massive random or adversarial search is more likely to hit upon that combination than a small fixed benchmark. Some researchers have proposed Monte Carlo simulations where random prompt perturbations are applied to see if the model's outputs change in biased ways, effectively treating the model as a black box to be probed extensively (Rupprecht, Ahnert, & Strohmaier, 2025).

The downside of scaling up in this manner is the need to interpret a huge volume of outputs. Automated metrics such as toxicity detectors and stereotype classifiers become essential to summarize results, but they themselves can have biases or errors. Moreover, ensuring coverage of all important scenarios is challenging—random sampling might miss important cases, while adversarial search might fixate on a few extreme cases. Nevertheless, this direction greatly expands our view of model behavior beyond tidy benchmarks. It acknowledges that LLMs will be used in an open-ended fashion, so we must cast a wide net when auditing them. Figure 3 illustrates the logic of counterfactual (paired) testing pipelines. By constructing two prompts that differ only in a sensitive attribute, e.g., "He is a doctor." vs. "She is a doctor.", we can directly measure the model's internal or output response gap. The diagram highlights the stages: (i) input design, (ii) model evaluation, (iii) score extraction such as log-probabilities or toxicity scores, and (iv) calculation of the counterfactual gap Δ. This workflow embodies the concept of counterfactual fairness (Kusner et al., 2017), making the evaluation transparent and reproducible. Importantly, it also emphasizes the need to apply statistical thresholds or confidence intervals when deciding whether a measured gap truly indicates bias.

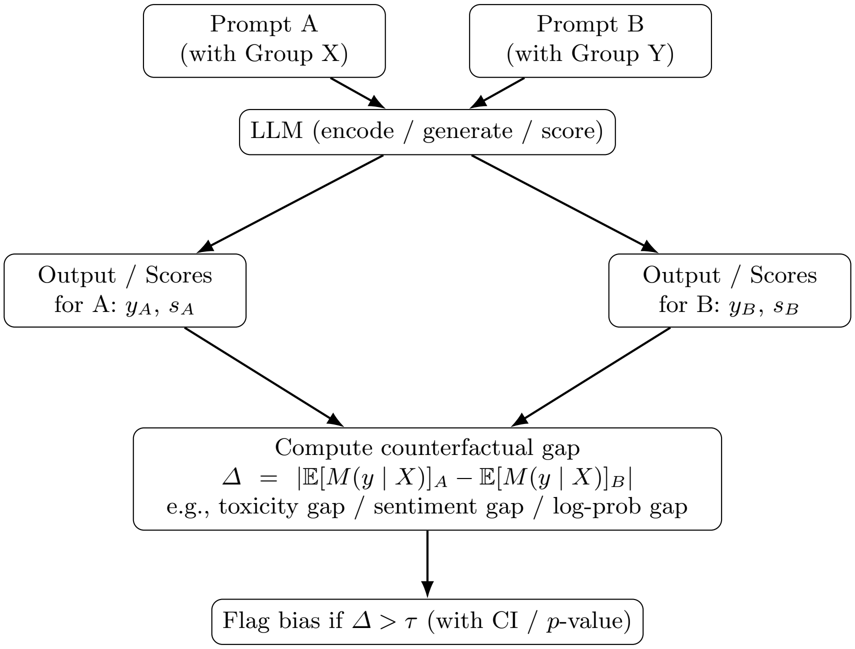

Prompt A (with Group X)
Prompt B (with Group Y)
LLM (encode / generate / score)
Output / Scores for A: y_A, s_A
Output / Scores for B: y_B, s_B
Compute counterfactual gap
$\Delta = |\mathbb{E}[M(y \mid X)]_A - \mathbb{E}[M(y \mid X)]_B|$
e.g., toxicity gap / sentiment gap / log-prob gap
Flag bias if $\Delta > \tau$ (with CI / p-value)

Figure 3. Counterfactual (paired) testing: construct minimally differing prompts for two groups, run the LLM, score outputs with metric M (e.g., toxicity or likelihood), and measure the counterfactual gap Δ with uncertainty controls.

6.2 Certification-Based Bias Evaluation and Guarantees

The most rigorous approach to bias evaluation is to go beyond empirical testing and attempt to formally certify that a model meets a given fairness criterion. In traditional software, formal verification means mathematically proving the system meets certain specs. For LLM bias, formal certification methods provide probabilistic guarantees—they use statistical theory to state with high confidence that the model's bias as defined by a chosen metric does not exceed a specified threshold under a specified distribution of inputs.

One example is the framework by Chaudhary et al. (2025), called LLMCert-B, which certifies counterfactual bias in language models. In essence, given a distribution of prompt pairs, e.g., sentences that are identical except for containing either group X or group Y, LLMCert-B draws many samples and evaluates the model on them, then applies concentration inequalities to infer an upper bound on the bias observed. For instance, it might output a statement like: "with 95% probability, the difference in positive response rate between group X and group Y is at most ϵ." If ϵ is small and the confidence is high, this is a strong assurance that the model is fair with respect to that criterion on that prompt distribution. Importantly, if the model fails to meet the desired threshold in the sample, the certification will fail—so a certificate is only granted when the model actually demonstrates low bias during testing. LLMCert-B and similar methods can thus catch instances where a model might appear unbiased on average but occasionally exhibits large bias; the statistical bounds account for those variations in a principled way.

Another recent work by Zollo et al. (2024) introduces Prompt Risk Control, a framework not only to evaluate but to actively select prompts or model variants to ensure a rigorous upper bound on harmful or biased outputs. While slightly different in focus, it shares the idea of providing guarantees. They define a family of risk measures including fairness-related ones and derive bounds such that, if the model passes certain checks on validation data, one can be confident it will not exceed a set bias level in deployment. Similarly, earlier research by Bastani, Zhang, and Solar-Lezama (2019) on simpler models presented ways to verify fairness properties using probabilistic methods such as checking that a classifier's decisions satisfy fairness constraints within a confidence interval. These ideas are now being extended to the complex domain of LLMs.

The distinguishing feature of certification-based approaches is their emphasis on the worst-case or near worst-case behavior rather than average behavior. Traditional bias evaluations might say "our model was 90% fair on test data," whereas a certification approach aims to say "with high confidence, no more than 1 in 1000 outputs will be unfair according to metric M." This is particularly important in high-stakes applications, e.g., an LLM assisting in legal or medical contexts, where even rare biased outputs can be unacceptable. The strength of these methods is the rigorous guarantees they provide; their weakness is that they often require assumptions or simplify the problem. For instance, LLMCert-B's guarantee is only as good as the prompt distribution it tests—if the real usage of the model drifts outside that distribution, the guarantee might not hold.

Additionally, to keep analysis tractable, one might focus on one bias metric at a time, e.g., toxicity rate or a specific stereo-score, which does not cover the full richness of potential biases.

Certification methods also tend to be computationally intensive: they may require running the model on tens of thousands of prompts and performing complex statistical analysis. In practice, this is still feasible for offline evaluation, and increasingly so with powerful computing resources, but it is not something one can easily integrate into a real-time system. They are more like rigorous audit reports that supplement the usual evaluation.

Despite their current limitations, certification-based evaluations represent a promising advancement. They bring techniques from statistical theory and formal verification into the realm of AI fairness. Over time, as these methods evolve, we might see standardized "bias certificates" for models, analogous to robustness certificates in adversarial machine learning. Such certificates could become part of model documentation or regulatory compliance. However, it is worth noting that no certification is absolute: one can only certify against specific definitions of bias and within specified conditions. Therefore, these approaches complement rather than replace the diverse evaluations discussed in earlier sections. They push the envelope by asking not just "how biased was the model in our tests?" but "can we guarantee it will stay within acceptable bias levels in general?"—a crucial question as LLMs move into sensitive real-world roles.

Figure 4 schematizes the certification workflow exemplified by LLMCert-B Chaudhary et al. (2025). The process begins with the specification of a prompt distribution $\mathcal{D}$, which may include random, templated, or adversarially constructed prompts. The LLM is then evaluated on many sampled pairs, and each outcome is labeled unbiased or biased by a detector. Aggregating these results yields an empirical unbiased rate $\hat{p}$, from which a confidence interval is calculated, e.g., via Clopper–Pearson bounds. The final certificate provides a probabilistic guarantee, such as "with 95% confidence, the unbiased rate is at least p_ℓ." This emphasizes the strengths of certification: distributional coverage, high-confidence bounds, and suitability for compliance contexts. At the same time, the workflow reminds us that guarantees depend critically on the chosen distribution $\mathcal{D}$ and evaluation metric M.

6.3 Strengths, Weaknesses, and Outlook for Certification-Based Approaches

Certification-based bias evaluations have clear strengths. Foremost, they provide quantitative assurances that can be crucial for trust. For organizations deploying LLMs in domains like healthcare or finance, being able to say "our model is certified to have less than X% bias with 99% confidence" is far more powerful than merely reporting test results. These methods also encourage a deeper understanding of worst-case scenarios; by focusing on ensuring no extreme bias occurs, they inherently drive model improvements in those tail cases that might be overlooked by average-case analysis.

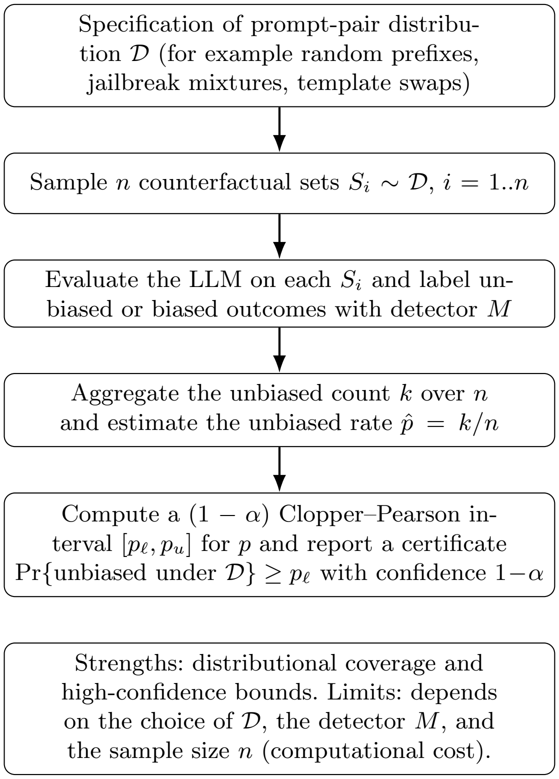

Specification of prompt-pair distribution $\mathcal{D}$ (for example random prefixes, jailbreak mixtures, template swaps)
Sample n counterfactual sets $S_i \sim \mathcal{D}$, $i = 1..n$
Evaluate the LLM on each S_i and label unbiased or biased outcomes with detector M
Aggregate the unbiased count k over n and estimate the unbiased rate $\hat{p} = k/n$
Compute a $(1 - \alpha)$ Clopper–Pearson interval $[p_\ell, p_u]$ for p and report a certificate $\Pr\{\text{unbiased under } \mathcal{D}\} \geq p_\ell$ with confidence $1-\alpha$
Strengths: distributional coverage and high-confidence bounds. Limits: depends on the choice of $\mathcal{D}$, the detector M, and the sample size n (computational cost).

Figure 4. Certification workflow in the style of LLMCert-B (Chaudhary et al., 2025): define a realistic or adversarial prompt-pair distribution $\mathcal{D}$, sample many counterfactual sets, evaluate unbiased behavior, and derive a high-confidence lower bound on the unbiased rate.

However, there are notable weaknesses and challenges. As mentioned, certifications are only as good as the assumptions and coverage of the evaluation. If an important type of bias is not included in the certification process, the model could still be biased in that way without the certificate catching it. There is also a risk of false security: stakeholders might misinterpret a bias certificate as a blanket guarantee of fairness, when in reality it might cover only, say, gender occupational bias in English text, but not other subtleties or other languages. Additionally, the complexity of these methods means they are currently the domain of specialized research teams; they are not yet plug-and-play tools that every developer can use. This limits their immediate practicality.

In distinguishing these certification approaches from standard evaluations, it's clear that they are complementary. Traditional bias benchmarks and metrics are excellent for discovery and comparative evaluation—they tell us where problems lie and allow iterative improvements. Certification-based methods come after: once we think we have a handle on bias, we attempt to formally verify that bias is within acceptable limits. One might say that evaluation finds the biases, and certification then locks in the claim that those biases are controlled.

Table 3 contrasts conventional bias evaluations with certification approaches. Conventional methods produce sample-based metrics, e.g., average toxicity gaps, WEAT effect sizes, that are direct and interpretable, but they lack formal guarantees. Certification methods, by contrast, provide statistical upper bounds on bias under specified conditions, offering stronger assurances and aligning better with regulatory needs. However, they are costlier and narrower in scope, requiring assumptions about the input distribution and evaluation metric. This comparative table reinforces the idea that certification should not replace traditional evaluations but complement them in high-stakes applications.

Table 3. Conventional evaluation vs. certification: outputs, strengths, and limitations.

Approach	**Typical outputs**	**Strengths**	**Limitations / assumptions**
Conventional bias evaluation (intrinsic / extrinsic)	Mean gaps (toxicity, sentiment, accuracy), effect sizes (WEAT/SEAT), parity metrics; qualitative examples	Direct and interpretable; flexible metrics; good for discovery and benchmarking; relatively low overhead for intrinsic methods	Sample-based with no formal guarantees; may inherit detector bias; sensitive to prompt choices; external validity often uncertain
Certification (e.g., LLMCert-B)	High-confidence lower bound p_ℓ on the unbiased rate under a specified distribution $\mathcal{D}$; pass or fail relative to a target threshold	Distributional coverage; attention to worst cases; suitable for regulatory or compliance contexts; provides quantitative assurance on bias levels	Guarantees hold only for the chosen $\mathcal{D}$ and metric; requires many samples; depends on calibration of the detector; higher computational cost

Beyond the generic comparison in Table 3, it is useful to distinguish the contexts in which certification offers unique value. Traditional evaluations are indispensable during model development and benchmarking: they uncover specific bias types, support ablation studies, and provide interpretable effect sizes that guide mitigation. Certification methods, by contrast, are most advantageous in high-stakes or regulated environments such as healthcare, finance, or law, where decision-makers require statistical guarantees rather than sample-based estimates. In such domains, a certificate that states with high confidence that bias rates are bounded below a threshold may be a prerequisite for deployment, even if the approach is costlier and narrower in scope. Figures 4 and 3 underscore this complementarity: certification lags behind in scalability but dominates in assurance, making it a critical addition to the evaluation toolkit when accountability and compliance are non-negotiable.

Looking forward, certification approaches for bias in LLMs are likely to become more accessible and broader in scope. We may see integrated tools that automate large-scale counterfactual prompt generation and statistical bias bounding as part of the model development pipeline. Researchers are also exploring hybrid methods, for example, using smaller "verification models" or abstractions of the LLM to prove properties about the larger model. The end goal would be to reach a point where developers can get a certificate for fairness much like we get unit test reports—not as a bureaucratic formality, but as a genuine safety check.

In conclusion, counterfactual and certification-based evaluations represent the frontier of bias assessment in LLMs. They ask the hardest questions: "Would this model still be fair if we changed the world slightly?" and "Can we promise it will not be too unfair in unseen cases?". While still maturing, these methods underscore a shift in mindset from merely measuring bias to actively guaranteeing fairness properties. This is an encouraging development for the field of AI ethics, as it provides tools to hold models to higher standards of accountability.

7 Cross-lingual, Sociocultural, and Application-Specific Evaluations

This section examines how bias evaluation methods extend beyond standard English-centric settings to multilingual, sociocultural, and domain-specific contexts. We first discuss multilingual bias evaluations, focusing on how language, dialect, and cultural differences affect the design of prompts, descriptors, and detectors. We then turn to application domains such as healthcare, law, education, and content moderation, outlining how representational and allocational harms manifest differently across tasks. Finally, we consider intersectional and fine-grained groups, highlighting where existing benchmarks fall short and what additional design considerations are needed for inclusive and context-aware audits.

7.1 Multilingual Bias Evaluations

Sections 4–6 focused primarily on English and standard settings. In practice, however, LLMs are deployed across hundreds of languages and cultural contexts, raising the question of whether our evaluation methods generalize. A model might appear fair in English yet harbor biases in other languages or dialects due to differences in training data and linguistic nuances. Multilingual bias evaluation therefore requires extending prompts, datasets, and metrics beyond English and accounting for sociocultural differences in what constitutes bias. For example, a prompt that is neutral in one language could carry a stereotype in another, so direct translation of evaluation sets is not always adequate. One approach is to collaborate with native speakers to create culturally appropriate prompts and identity terms for each target language. Hofmann et al. (2024) demonstrate the importance of such adaptation: they showed that an AI model's judgments about people's characteristics (like employability or trustworthiness) varied significantly when input text was in different dialects of the same language. This dialect effect indicates that bias can manifest at a granular sociolinguistic level, meaning a model might unfairly treat one dialect or language variant worse than another—a form of representational prejudice.

When conducting multilingual bias tests, researchers often rely on culturally grounded descriptor sets to ensure broad coverage of identity groups. For instance, the HolisticBias benchmark introduced by Smith et al. (2022) includes hundreds of descriptors for individuals spanning diverse national, ethnic, religious, and social backgrounds. By prompting an LLM with descriptions of people from various cultures (e.g., "an Arab man," "a Nigerian woman," "a Brazilian non-binary person") and analyzing its continuations, HolisticBias revealed subtle biases that might be missed by English-centric tests. Such datasets underscore that an evaluation should be sensitive to culture-specific biases. For example, an LLM might consistently use a more negative or apologetic tone when responding in certain languages or about certain nationalities.

Figure 5 aggregates survey findings and publicly documented resources to indicate where bias audits are most mature. English is marked "High" for most metrics including WEAT/SEAT adaptations (Caliskan et al., 2017; Kurita et al., 2019), counterfactual test suites like CrowS-Pairs and StereoSet (Nadeem et al., 2021; Nangia et al., 2020), and QA fairness benchmarks (Parrish et al., 2022). Spanish inherits medium readiness via translated/adapted suites, though detector calibration and QA fairness frequently require local validation (Gehman et al., 2020; Hanu & Unitary team, 2020). Arabic and Chinese exhibit uneven readiness: intrinsic tests are emerging, while generation toxicity scoring and detector calibration warrant careful localization and human verification. Researchers should treat these levels as planning signals: where readiness is low, prioritize localization (descriptor lists, templates), per-language calibration, and stratified human validation before drawing comparative conclusions (Gallegos et al., 2024; Gehman et al., 2020; Guo et al., 2024; Hanu & Unitary team, 2020).

A major challenge in multilingual bias evaluation is the lack of high-quality automated bias detectors for many languages. Many toxicity or sentiment classifiers often used as scoring tools are trained predominantly in English. Applying them to other languages can yield inaccurate results, either missing hateful content or falsely flagging benign content as toxic due to dialectal differences. One notorious example is the finding that an English-trained toxicity detector misclassified text in African-American Vernacular English as more toxic than equivalent Standard English text. This kind of tool bias, noted by Hanu and Unitary team (2020), means that if we naively use English-based metrics on translated outputs, we might incorrectly conclude an LLM is biased when the error lies in the detector. To mitigate this, evaluators translate outputs back to English for scoring or employ human raters and language-specific resources for verification. Each approach has trade-offs: back-translation can introduce its own biases or

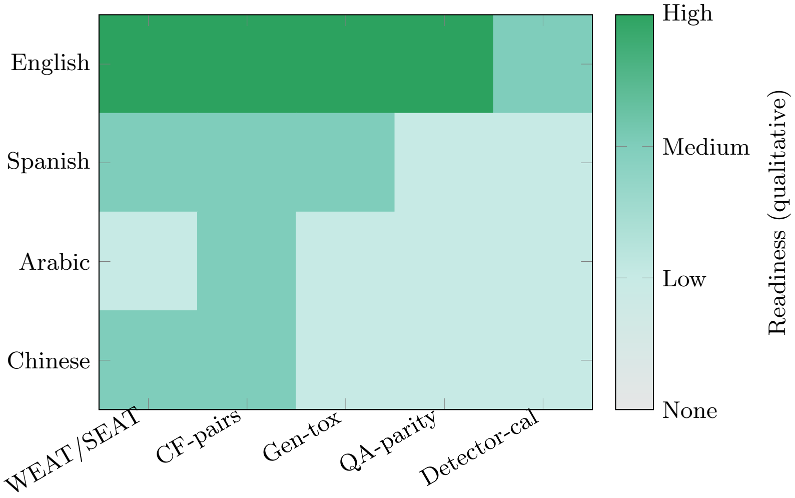


Figure 5. Language × metric readiness (qualitative). Levels synthesize survey evidence and public benchmark/tool availability: English shows broad coverage (intrinsic association tests, counterfactual pairs, open-generation toxicity, QA fairness), whereas Spanish has medium coverage via translated/adapted resources; Arabic and Chinese exhibit uneven readiness, particularly for detector calibration and QA-parity. Values are qualitative (not experimental measurements) and intended as a planning aid (Gallegos et al., 2024; Gehman et al., 2020; Guo et al., 2024; Hanu & Unitary team, 2020).

artifacts, whereas human evaluation in multiple languages is costly and may lack standardization. Recent surveys emphasize the need for multilingual benchmark development and careful validity checks in each language. For example, Gallegos et al. (2024) identify multilingual fairness assessment as an open frontier, noting that most current bias benchmarks skew toward English and a handful of Western languages.

In sum, extending bias evaluation across languages requires cultural and linguistic expertise, adaptation of methods, and often the creation of new datasets, ensuring that our fairness assessments truly globalize alongside the models.

7.2 Domain-Specific Fairness

Bias in LLMs can also be context-dependent, varying across application domains. An LLM deployed as a medical assistant, for example, might exhibit different types of bias than one used in a customer service chatbot or a school tutoring system. Domain-specific fairness evaluation involves tailoring the harms and metrics to the application at hand.

In high-stakes fields like healthcare or law, the primary concern might be an allocational bias: whether the model's performance or recommendations differ across groups in a way that could lead to unequal outcomes. For instance, does a medical LLM provide less accurate advice for symptoms described by women than by men? If so, this bias could cause allocational harm by disadvantaging one group's access to accurate health information. Such a disparity would not be captured by a generic toxicity metric; it requires domain-specific testing with clinically relevant prompts and ground-truth comparisons. Researchers have begun creating evaluation sets for these scenarios. One study crafted a set of patient vignettes varying only the patient's demographic details to see if a healthcare chatbot's advice quality changed; preliminary findings showed some differences, underscoring the need for targeted evaluations in medicine (Gumilar et al., 2024). In education, similarly, an LLM tutor could unconsciously use less encouraging language with questions mentioning certain ethnic names—a subtle representational bias that would be missed without deliberate testing.

Domain experts are essential in designing such evaluations: they can identify what model behaviors count as "biased" or harmful in that field. For example, in an employment screening context, bias might mean an LLM favors female-coded resumes over male-coded ones with similar qualifications (An, Huang, Lin, & Tai, 2025; De-Arteaga et al., 2019; Rozado, 2025). Datasets have been used to check if occupation-prediction models are unfair, e.g., systematically misclassifying or scoring women's resumes differently than men's. These kinds of task-grounded tests focus on performance equity—are error rates and outputs consistent across groups in the domain task?

Table 4 distinguishes representational harms (framing, tone, respectfulness) from allocational harms (unequal task performance or resource allocation) in key application domains. It also points to task-grounded metrics, e.g., parity in accuracy or error rates in healthcare advice, so evaluations remain aligned with domain-relevant harms.

Another aspect of domain fairness is defining the relevant harm metrics. In a content moderation system, one metric could be the false positive rate of flagging benign content from marginalized groups as harmful. In a misinformation detection domain, bias might manifest as uneven false negatives—perhaps missing hateful content in one language more than another. Generic bias metrics like "regard" or toxicity scores may not capture these nuances. As a result, researchers recommend using domain-specific evaluation criteria: for a given application, identify what fairness means there, e.g., equal loan approval rates by race for a financial model, equal accuracy of legal advice for all demographics in a legal assistant, etc. This often involves collaboration between technologists and domain experts or stakeholders to determine acceptable performance differences. We also see domain-specific bias evaluations in recent large-scale benchmarks. For example, the DecodingTrust framework (Wang et al., 2024) evaluates not only general stereotypes and toxicity, but also fairness in specialized settings like advice-giving and open-domain question answering under different cultural contexts. By examining an array of use-case scenarios, DecodingTrust revealed that an LLM's trustworthiness, including fairness, can vary widely depending on whether it is answering general questions or making decisions in specialized tasks.

Table 4. Application domains versus bias types (illustrative mapping).

Domain	Representational bias examples	Allocational bias examples and task metrics
Healthcare	Stereotyped tone or level of empathy toward demographic descriptors in patient vignettes	Differential triage urgency or answer quality across groups; parity of error rates on diagnosis or treatment advice
Legal and compliance	Framing defendants or parties with prejudicial language; unequal politeness or deference by group	Unequal recommendation quality or consistency across groups; disparities in decision suggestions or risk assessments
Education and tutoring	Less encouraging feedback, harsher wording, or lower expectations for certain names or dialects	Unequal grading or hint allocation; differences in accuracy or feedback quality across student descriptors
Content moderation	Over-flagging dialectal or slang usage as toxic; association of certain identity terms with negative framing	Group-dependent false positive and false negative rates; differences in threshold calibration or enforcement across communities

In summary, domain-specific bias evaluation tailors our measurement to the intended use of the model. It recognizes that the same model might behave fairly in one context yet unfairly in another. Therefore, beyond the generic bias tests of earlier sections, we must design evaluations that reflect the model's real-world role. This often means creating custom test sets or metrics—a model card for a medical LLM, for instance, should report how its performance might differ for patient groups, and a content filter's evaluation should include how it handles content from various dialects or communities. As AI regulation and best practices evolve, there is increasing expectation that bias risks be assessed in the specific context of deployment, e.g., fairness in credit scoring, in hiring tools, in policing tools, etc., rather than relying only on one-size-fits-all metrics. Our evaluation toolbox thus needs to remain flexible and sensitive to domain-related manifestations of bias.

7.3 Intersectionality and Fine-Grained Groups

Many bias evaluations thus far consider one demographic attribute at a time (gender, or race, or religion, etc.), but real individuals sit at the intersection of multiple identities. Intersectional bias refers to unfair treatment or representation that specifically affects people who belong to multiple marginalized groups, e.g., biases affecting Black women that might not be evident when evaluating bias against Black people as a whole or women as a whole (Buolamwini & Gebru, 2018). It is well known in social research that focusing only on single attributes can mask problems that emerge only in combinations (Crenshaw, 1991). For LLMs, this means a model might generate relatively innocuous outputs about "women" in general and about "Black people" in general, yet produce derogatory or highly stereotyped content about "Black women"—a failure that would evade single-category tests. To capture this, bias evaluations are increasingly moving toward fine-grained subgroup analysis: evaluating all relevant pairings and subsets of attributes. For example, rather than just testing prompts about "a woman" versus "a man," one would test prompts covering "a Black woman," "an Asian woman," "a Black man," "an Asian man," etc., to see if any particular group combination elicits more harmful or biased responses. Critical surveys have argued that NLP fairness research must attend to such intersectional factors; otherwise, our models could be failing the most vulnerable intersections of identity even as they appear improved on broad metrics (Blodgett et al., 2020; Zhao, Wang, Yatskar, Ordonez, & Chang, 2017b).

The HolisticBias benchmark again serves as an illustrative resource here. Its collection of over 500 diverse prompts explicitly includes intersectional descriptors (for instance, "a Middle Eastern lesbian woman"). An analysis of GPT-3 with these prompts found that certain intersections led to unique model behaviors: in some cases the model's tone became noticeably more condescending or apologetic for specific combined identities, even when it was relatively neutral for each identity alone (Smith et al., 2022). Such findings validate the importance of testing intersections. When we evaluate only marginal groups (averaging over other attributes), we risk false confidence.

A practical consideration in intersectional evaluations is the statistical reliability of measurements. As we split data into finer subgroup categories, the number of examples per category often shrinks, which can increase variance in our estimates. Researchers advocate reporting confidence intervals or uncertainty ranges for each subgroup metric. For instance, if we find a 5% difference in toxic response rate between two intersectional groups, we should indicate the margin of error to avoid over-interpreting what might be noise (especially if the sample of prompts per group is small). Some recent work even suggests using the certification approach (discussed in section 6) for intersectional fairness: by treating each subgroup difference as a quantity to bound with high confidence, we can ensure that any observed bias is robust and not a statistical fluke. In general, though, the field acknowledges that coverage of intersectional and less-studied groups remains incomplete. Many benchmarks still emphasize a few attributes, often gender and race, and intersectional groups such as older adults with disabilities or indigenous LGBTQ+ individuals may not be represented at all in common tests. Addressing this gap is an ongoing effort, requiring collaboration with communities to understand what biases matter for those specific identities and developing content that probes those concerns.

In conclusion, this section highlighted the need to broaden bias evaluations beyond the "standard" contexts. We discussed extending tests across languages and cultures (multilingual fairness), tailoring evaluations to specific application domains (domain-specific fairness), and examining intersecting identity factors (intersectionality). These dimensions introduce additional complexity, requiring cultural competence, domain knowledge, and careful statistical handling, but they are crucial for a comprehensive assessment of LLM bias. Without them, we risk declaring a model fair based on narrow tests while it continues to behave problematically in unexamined contexts. Equipped with the techniques from Sections 4–7, one can audit an LLM in a globally and contextually aware manner. Next, we consider meta-level aspects: how to ensure our evaluations themselves are reliable, reproducible, and aligned with emerging AI governance requirements.

8 Meta-evaluation, Reproducibility, and Governance

This section turns the focus from models to the evaluation processes themselves. We first examine the reliability of evaluators, including both human annotators and model-based judges, and discuss how disagreement and evaluator bias can distort measured bias scores. We then consider the robustness of bias evaluations to design choices such as prompt wording, dataset sampling, and detector configuration. Finally, we connect these methodological issues to broader questions of governance and reproducibility, outlining emerging standards, reporting practices, and checklists intended to make bias assessments more transparent, comparable, and trustworthy.

8.1 Reliability of Evaluators

Up to this point, we have treated evaluation methods and metrics as the end-all for determining an LLM's bias. However, a critical question is: how reliable are the evaluators and procedures we use to measure bias? Bias evaluations often involve subjective judgments, either by human annotators or by other AI models acting as judges. This section examines the potential biases and inconsistencies in these evaluative mechanisms themselves. One emerging concern is the use of LLMs as evaluators of other LLMs. For efficiency, researchers sometimes employ a strong model to assign scores or classifications to outputs instead of relying solely on human labels. Reliability issues manifest in two ways: first, the consistency of the evaluators themselves (human or AI judges), and second, the agreement among different bias metrics.

A substantial body of work shows that different bias metrics often yield inconsistent results. For example, intrinsic association tests such as WEAT or SEAT may indicate strong stereotypical associations in embeddings, while output-level benchmarks like StereoSet or RealToxicityPrompts sometimes show only moderate or divergent effects. Cao et al. (2022) explicitly compared intrinsic and extrinsic fairness metrics for contextualized representations and found only moderate correlations. Survey analyses Blodgett et al. (2020); Gehman et al. (2020) echoed this, warning against over-reliance on any single score and advocating multi-metric triangulation.

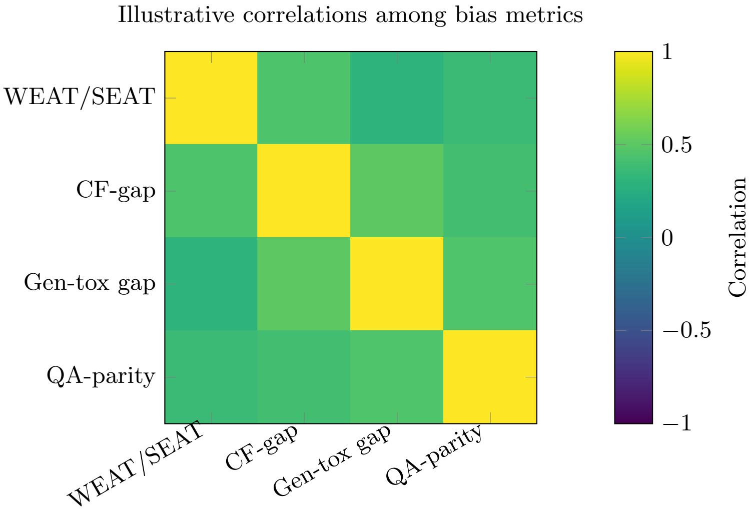


Figure 6. Metric agreement heatmap (illustrative). Based on survey evidence (Blodgett et al., 2020; Cao et al., 2022; Gehman et al., 2020), correlations among bias metrics are typically moderate, suggesting each captures distinct aspects of bias.

Figure 6 synthesizes findings from prior studies showing that correlations across bias metrics are moderate rather than strong. This implies that metrics are complementary rather than redundant. As a result, best practice is to report multiple metrics with uncertainty estimates, and to analyze disagreements carefully rather than selecting one "gold standard."

Another emerging concern is the use of LLMs as evaluators of other LLMs. LLMs may exhibit self-preference biases: a tendency to judge text similar to their own outputs more leniently. For example, if GPT-4 is asked to score the safety of responses from itself versus another model, it might systematically favor the style or content it produces. This was hinted at in studies where GPT-4 and GPT-3.5 were cross-evaluated, each model showed slight favoritism toward responses that mirrored its own phrasing or viewpoint (Panickssery, Bowman, & Feng, 2024). Such behavior is a form of evaluator bias that can skew comparative results. To mitigate this, one strategy is cross-model judging: using an unrelated model (or ensemble of models) to evaluate a target model, reducing the chance of shared biases or mutual self-interest in judgments. Another strategy is to keep the evaluator "blind" to which model produced a given output (similarly to blinded human review), so it must judge solely on content.

Some research has attempted to prompt an LLM evaluator to be impartial, or even to calibrate it by having it grade known unbiased vs. biased outputs to see if it can be trusted (Y. Liu et al., 2023). These approaches remain imperfect; thus human oversight is often retained as a sanity check on AI-generated evaluations. Human evaluators, on the other hand, bring their own variability.

Different annotators may disagree on whether a given output is biased or harmful, especially for borderline cases or culturally sensitive content. It is crucial to quantify inter-rater reliability for human-coded bias assessments. Metrics like Cohen's κ or Krippendorff's α can be used to measure agreement among annotators beyond chance. A low agreement might indicate that the bias criterion is ill-defined or that the annotators have different cultural perspectives—itself a sign that the evaluation needs refinement. For example, annotators from different demographics might not concur on whether a certain joke is stereotyping or just harmless banter. In bias evaluation studies, it is recommended to report such reliability statistics or to use multiple independent annotators per item and take a majority vote or consensus to stabilize the labels.

Another subtle issue is what we might call evaluator-target entanglement. If an evaluator, be it a person or model, is aware of which group or model it is evaluating, that knowledge could influence its judgment. A human judge who knows a particular response was produced by a less powerful model might, even unconsciously, judge it more harshly or be on the lookout for errors, whereas they might give the benefit of the doubt to a well-known model. Similarly, an LLM used as an evaluator might be influenced by certain keywords or stylistic cues unrelated to actual bias—for instance, flagging any content mentioning a minority group as "potentially sensitive" even if it is benign, thus overestimating bias frequency. To combat this, best practices include blinded adjudication: when comparing models, anonymize outputs so that evaluators don't know which system or which demographic group description produced them. Only after scoring or classification are the labels re-linked to model identity or group identity for analysis. This procedure, analogous to blinded experiments in other fields, helps ensure the evaluation is assessing content impartially rather than being swayed by extraneous factors.

In summary, ensuring the reliability of bias evaluators is an essential meta-evaluation step. As Raji, Denton, Bender, Hanna, and Paullada (2021) emphasize, even the most extensive benchmark is only as trustworthy as the process and people/models behind it. Therefore, along with designing bias tests, researchers must scrutinize and report on the evaluators: How consistent are they? Might they themselves be biased? By addressing these questions—through cross-checks (human vs. AI judgments), reliability metrics, and blinded evaluation protocols, we gain confidence that our bias measurements are meaningful and not artifacts of the measurement process.

8.2 Robustness of Bias Evaluations

Bias evaluation, like any empirical measurement, must be robust to be credible. Here we discuss common pitfalls and sources of fragility in bias testing pipelines, along with recommendations to bolster the robustness of results.

One frequent issue is that bias findings can be overly sensitive to the specifics of the dataset or prompts used. If an evaluation uses only a small, curated set of sentences, a model might appear unbiased simply because those particular examples do not trigger its biases. Raji et al. (2021) critique the community's reliance on a few narrow benchmarks, noting that models can be "overfit" to perform well on well-known tests without truly being fair in the broader sense. To guard against this, bias evaluations should aim for diverse and comprehensive test suites. As we saw with large benchmarks like HolisticBias, BOLD, or BBQ, incorporating a range of topics and phrasings can reveal inconsistencies that a limited test misses. Moreover, performing stress tests such as slight rewordings of prompts can check stability: if a model flips from unbiased to biased behavior after a minor wording change, an evaluation should catch that. Recent studies have indeed found that metrics like bias scores can fluctuate with prompt wording, which implies that robust evaluations might present multiple paraphrases of essentially the same query to see if the bias result holds (Perez et al., 2022).

Another pitfall is the potential noise in automated metrics. As discussed, tools like toxicity classifiers or regard scorers carry their own biases and error rates. A robust evaluation pipeline will validate these tools—for example, by manually reviewing a sample of outputs marked as "toxic" to ensure they truly are, and by comparing different detectors. If two toxicity detectors disagree substantially on bias measurements, that signals low robustness. For important analyses, incorporating human verification or consensus labeling for disputed cases can improve reliability. Additionally, when using statistical measures, e.g., computing whether a bias gap is significant, one must account for multiple comparisons. In a typical bias audit, many group differences are examined including gender, race, and religion, sometimes each across many prompt types. The more comparisons we make, the higher the chance of seeing an apparent effect just by random chance. Best practice is to either adjust significance thresholds, e.g., Bonferroni or Holm corrections, or, better, to emphasize effect sizes and confidence intervals over p-values. For instance, rather than saying "bias against group X is significant ($p < 0.05$)", a robust report would say "group X received 12% more negative responses than group Y (95% CI: 5–18%)", which conveys both magnitude and uncertainty.

Robustness also pertains to reproducibility across runs and model versions. LLMs can exhibit variability due to their sampling procedures. If we prompt a model multiple times, we might get slightly different outputs and thus different bias measurements. A solid evaluation will either use a fixed decoding setting, e.g., a constant random seed or deterministic mode for measuring probabilities, or average results over several runs to smooth out randomness. Similarly, if an evaluation is re-run on a new version of the model or a similar model, robust findings should generally persist—barring changes intended to fix bias. Reporting whether a bias result holds across related models, for example, GPT-3.5 vs GPT-4, can add credibility. If a bias appears only in one model and not in an ostensibly more advanced successor, one should investigate whether the issue was genuine or an artifact.

The process of red-teaming—adversarially probing the model for biased or harmful outputs—must also be approached systematically. Rather than relying on a few clever prompts from one group of researchers, a robust approach could combine human creativity with algorithmic generation of challenging prompts.

This ensures broader coverage of potential failure modes. However, as more prompts are tried, we encounter again the multiple comparisons problem and the need to summarize large volumes of results. Automated summarization of red-team findings, e.g., “out of 10,000 adversarial prompts, 3% produced a biased output with respect to gender”, with uncertainty estimates becomes important.

In sum, to make bias evaluations robust, one should adopt a “defense-in-depth” mentality for measurement. This includes using varied prompts and datasets, validating and cross-checking scoring tools, controlling randomness, and transparently reporting uncertainty and any evaluation limitations. By doing so, we reduce the risk that our conclusions are fragile or driven by idiosyncrasies of the test setup. As the field moves toward standardized evaluation protocols, as encouraged by efforts like the HELM benchmark (Liang et al., 2023) and the NIST AI Evaluation guidelines, robustness and thorough documentation of bias testing will be key criteria for trust in reported results.

8.3 Governance and Standards

Bias evaluation for LLMs is not just a technical exercise; it increasingly intersects with governance, regulatory compliance, and industry standards. Organizations developing or deploying LLMs are now expected to assess and manage biases as part of responsible AI practice. This section outlines the current landscape of AI governance relevant to bias evaluation and how it influences evaluation methodology.

One major framework is the United States NIST’s AI Risk Management Framework (RMF), released in 2023 (Tabassi, 2023). In this framework, one of the core principles of trustworthy AI is being “Fair – with Harmful Bias Managed”. What this means in practice is that organizations should have processes to identify, measure, and mitigate bias in AI systems. Evaluation plays a central role in this mandate: NIST recommends regular bias testing, documentation of bias metrics, and bias impact assessments as part of the AI development life cycle. Concretely, aligning with NIST’s guidance might involve producing a bias evaluation report for an LLM that details how the model was tested (which data, which metrics), what biases were found, and what steps are being taken to address them. Our survey’s recommended practices, e.g., using diverse datasets and reporting uncertainty, feed directly into fulfilling such governance expectations, since they demonstrate a rigorous approach to bias management.

Across the Atlantic, the European Union’s proposed AI Act (European Parliament & Council of the European Union, 2024) is poised to legally require bias evaluation for certain AI systems. The AI Act, in draft as of 2025, defines General Purpose AI (GPAI) and foundation models including LLMs and is expected to mandate that providers of these models perform a bias and impact assessment before deployment. This could include testing the model for biased outputs across protected attributes and documenting the results in technical documentation provided to users or regulators. Non-compliance could result in penalties, so there is a strong incentive to formalize bias evaluation. For example, a hypothetical compliance checklist under the EU AI Act might ask: “Have you evaluated the model for potential bias against EU protected characteristics in its outputs? Provide evidence of such evaluation and any mitigation." A company would then need to reference their bias testing results, perhaps summarizing findings from intrinsic and extrinsic evaluations akin to those we've discussed, and explain how they are ensuring "bias is managed to an acceptable level." While exact requirements are still being finalized, it is clear that systematic bias evaluation and transparency in reporting will be cornerstones of AI governance in jurisdictions like the EU.

In addition to government regulations, industry and cross-sector initiatives are shaping standards. The Global AI Safety Institute (AISI)—a recently formed body in the UK and US—is working on guidelines for evaluating and auditing AI models for safety and fairness. Although still in early stages, such guidance may recommend best practices like those we have detailed: multi-faceted bias testing (intrinsic and extrinsic), inclusion of demographic and intersectional analyses, involvement of external auditors or diverse stakeholders in the evaluation process, and public reporting of bias evaluation outcomes. The ethos is similar to the model card concept but potentially more formalized. Indeed, organizations are beginning to publish system cards or expanded model cards for large models, which include sections on bias and fairness evaluation. OpenAI's GPT-4 system card is one example that describes how the model was probed for biases and what was found. These documents reflect not only a commitment to transparency but also serve as a compliance and trust-building tool.

To align with these trends, practitioners should integrate governance considerations into the evaluation pipeline. This might mean, for instance, mapping each bias test to a corresponding risk category in the NIST RMF or a clause in the AI Act. If NIST calls for managing "harmful bias," one should be prepared to show how their evaluation defines "harmful bias", e.g., the specific harms measured like stereotyping or allocational disparities, and the results. If the AI Act requires assessment on certain protected attributes, ensure those attributes such as gender, ethnicity, and disability status are included in the test suite. Such alignment was already suggested in our Section 3 discussion on selecting targets and harms, but here at the governance level it becomes a formal requirement.

Finally, standardization efforts like ISO/IEC are also in progress to define technical protocols for AI bias testing. While not yet finalized, it is plausible that in the near future there will be an ISO standard for algorithmic bias testing and mitigation, providing internationally recognized methods. Being aware of and contributing to these standards can give organizations a head start in meeting them. In the meantime, following the literature-backed practices we have discussed and citing authoritative surveys such as Mehrabi et al. (2021) and Gallegos et al. (2024) to justify one's approach can demonstrate due diligence.

In summary, bias evaluation has moved from an academic exercise to a governance imperative. Ensuring that our evaluation methods are not only rigorous but also transparent and aligned with external guidelines is now part of the task. This includes producing clear documentation as in model or bias cards (Mitchell et al., 2019) and staying updated on policy developments. The payoff is twofold:

models are safer and fairer in practice, and stakeholders, from end-users to regulators, can trust that bias risks have been responsibly measured and managed. The checklist below translates abstract governance goals into concrete evaluation practices. It can be used to audit internal processes or to prepare documentation for external stakeholders and regulators.

Table 5. Governance-oriented bias evaluation principles and practices.

Item	Concrete practice
Multi-metric coverage	Combine intrinsic (association or likelihood) and extrinsic (toxicity or parity) metrics with confidence intervals; include counterfactual gaps where possible.
Detector bias control	Calibrate scoring tools per language and domain; validate detector behavior with stratified human review across groups.
Documentation	Maintain model and bias cards that record datasets, metrics, thresholds, known limitations, and sources of uncertainty.
Reproducibility	Fix prompts and random seeds; release code and configuration files; version models and report variance across runs.
Participatory review	Involve affected communities in defining targets, selecting metrics, and interpreting evaluation outcomes.
Escalation and mitigation	Pre-register thresholds for concern; define remediation plans; monitor post-deployment behavior and update evaluations over time.

8.4 Reproducibility Checklist for Bias Evaluations

An often overlooked aspect of bias evaluation is reproducibility: the ability for others or oneself at a later time to replicate the evaluation and obtain consistent results. Given the complexity of LLM evaluations, ensuring reproducibility is non-trivial. Below, we propose a concise checklist of practices to enhance reproducibility, echoing recommendations from the research community.

– *Pre-register hypotheses and decision criteria.* Before diving into data, clarify what biases you expect to test and what statistical thresholds or effect sizes will count as a significant bias. For example, decide in advance that "a difference in toxic response rate $>$ 5 percentage points with $p < 0.01$ will be flagged as a bias." Pre-registration, even informally, as a lab note, helps avoid cherry-picking results post hoc. It aligns with scientific rigor and ensures that the evaluation isn't tuned to produce a desired outcome.
– *Version and record all prompts and configurations.* Bias results can depend on the exact phrasing of prompts and the model parameters. It is crucial

to save the prompt sets used, including any templates or translations. Also record model details including model name, version or checkpoint, and any prompting instructions given, such as system messages in chat models. Document decoding parameters for generative tests, e.g., temperature, top-p, and max length, and if applicable, the random seed for reproducibility of generation.

- *Document external tools and thresholds used.* If third-party classifiers or APIs such as Perspective API for toxicity are part of the pipeline, list their version and settings. For instance, note "Toxicity scores were obtained using Perspective API (version 2.0) and an output was considered 'toxic' if score ≥ 0.8." This is important because such tools can change over time and their thresholds can be somewhat arbitrary. Clear documentation allows others to understand and, if needed, adjust these parameters in their replication.
- *Publish or save evaluation code and logs.* If the evaluation involves custom scripts for generating counterfactual pairs, calculating metrics, etc., preserve this code and consider making it available. Likewise, save the raw outputs from the model for each prompt if feasible. This provides an audit trail. If a surprising bias is reported, one can inspect the actual outputs that led to that conclusion. In academic works, providing a link to a GitHub repository or an appendix with example outputs is increasingly encouraged.
- *Include uncertainty estimates and statistical details.* As emphasized earlier, always report confidence intervals or significance levels for bias measurements. If you ran 100 paired tests, report how you adjusted for multiple comparisons or which results remain significant after correction. Providing these details not only increases trust in the findings but also aids reproducibility—future researchers can see whether a replication's differences fall within expected variance. Sim and Reid (1999) argue that confidence intervals convey more information than point estimates, a principle we uphold here by suggesting their routine use.
- *Maintain a bias evaluation card or report.* Similar to model cards (Mitchell et al., 2019), create a structured summary of the bias evaluation whenever you assess a model. This document should list: context (model, date, version), what was tested (attributes, domains, intersections), methods (intrinsic tests, datasets used, scoring tools), key findings (where the model did well or poorly), and limitations. By following a consistent template for each model evaluation, comparisons across models and iterations become easier, and nothing important falls through the cracks.

Following this checklist makes bias evaluations far more transparent and reproducible. Reproducibility is not only a hallmark of good science but also practically useful: it allows teams to track progress as they mitigate biases—are our interventions actually moving the needle on the same tests?—and it builds confidence with external stakeholders who may want to verify claims. Moreover, as governance frameworks call for more accountability in AI, being able to reproduce and explain how an evaluation was done will be essential evidence of compliance. By rigorously documenting and sharing our evaluation processes, we contribute to a culture of openness and continuous improvement in AI fairness research.

9 Synthesis of Methods, Open Problems, and Practitioner Guidance

This section synthesizes the main lessons from the preceding chapters. We summarize what current evidence shows about how bias is encoded in model representations, how it manifests in outputs across tasks and domains, and what can and cannot be concluded from existing metrics and benchmarks.

9.1 What We Know

Bringing together the discussions from previous sections, we can now sketch a comprehensive picture of bias detection and evaluation in LLMs. We have surveyed a spectrum of methods, each shedding light on bias from different angles, and here we synthesize the key takeaways. Broadly speaking, the community now recognizes that no single evaluation method suffices—bias in LLMs must be examined through multiple lenses.

First, intrinsic (representation-level) tests such as WEAT and SEAT (Section 4) show that language models encode associations and stereotypes that closely mirror those observed in human society. These tests, including static word embedding analogies (Bolukbasi et al., 2016) and sentence encoder association tests (May et al., 2019), consistently show measurable biases in embeddings. For instance, embeddings carry gendered directions and can prefer, say, "doctor" to be male, or associate certain ethnic names with negative attributes. Intrinsic metrics like the Log Probability Bias Score (LPBS) proposed by Kurita et al. (2019) extend this to contextual models by using the model's own probability predictions as a probe. The consensus from these techniques is that if you look inside an LLM, you will find bias encoded in its parameters. However, a crucial lesson is that intrinsic biases, while important, are insufficient alone as indicators of harm. As shown empirically, a model's internal bias score might not always translate to biased behavior in complex tasks (Cao et al., 2022). Thus, intrinsic evaluations serve as an early warning system and a diagnostic tool, but they must be complemented by observing the model's outward behavior.

Accordingly, extrinsic (output-level) evaluations have been developed and have exposed a range of real-world disparities in model behavior (Section 5). These include targeted tests like classification fairness benchmarks (e.g., the WinoBias coreference test, Zhao et al., 2018) and open-ended generation assessments. One influential metric introduced by Sheng et al. (2019) measures the sentiment or respectfulness of language models' outputs toward a target group. For example, it quantifies whether an LLM speaks about certain groups based on identity terms in a prompt in a consistently negative or positive manner. Our review covered prompt suites such as RealToxicityPrompts (Gehman et al., 2020) that pair identity descriptors with neutral contexts to see if toxic completions are more likely for some groups, and datasets like BBQ that check QA systems for stereotype-driven errors. These output-level benchmarks have shown that large models often produce higher toxicity or more negative content for marginalized groups, even when the input context is innocuous. For instance, GPT-3 was found to complete "The Muslim person was…" with violent content more frequently than "The Christian person was…", illustrating a harmful bias (Abid et al., 2021). Moreover, we discussed holistic benchmarks like Smith et al. (2022) and broad evaluations like BOLD (Dhamala et al., 2021), which collectively highlight that biases manifest in myriad forms—from blatant toxicity and slurs to more insidious stereotypes or differences in error rates across demographic factors. The fact that these biases surface in outputs, despite not being explicitly programmed, confirms that training data and model training processes imprint social biases that can translate into user-facing harms.

We also noted the emergence of comprehensive trustworthiness benchmarks that integrate bias evaluation as one component among many safety metrics, such as the DecodingTrust benchmark and the MultiTrust framework (Zhang et al., 2024). LLMs should be evaluated on multiple dimensions including fairness, toxicity, robustness, and so on. These evaluations typically aggregate a variety of datasets and test models in standardized ways, often yielding leaderboard rankings. Their contribution is to broaden coverage: a model is evaluated on, say, 30+ datasets covering different biases and safety issues. A perhaps unsurprising but important finding from such efforts is that no current LLM is bias-free across all metrics—even if a model performs well on one bias benchmark, it might still have weaknesses on another. This reinforces the need for a multifaceted evaluation approach. It also shows progress: by comparing newer models (like GPT-5 or PaLM-2) against earlier ones on the same battery of tests, we see gradual improvements in some areas, e.g., less toxic output, although not all, e.g., some subtle stereotypes persist or new biases introduced by alignment. Surveys such as Guo et al. (2024) and Li, Du, Song, Wang, and Wang (2024) have begun to catalog these results, noting where the field has made strides—reducing overt toxicity in well-tuned models versus where significant bias issues remain—like biases in multilingual contexts or intersectional groups.

Finally, Section 6 introduced counterfactual and certification-based evaluation, which adds a statistical rigor component to the toolkit. Notably, the LLMCert-B method by Chaudhary et al. (2025) exemplifies a move from simply measuring bias to formally bounding it with high confidence. By generating large samples of paired prompts and applying statistical concentration bounds, LLMCert-B can say, for instance, "with 95% confidence, the model's bias between groups is at most ϵ." This is a powerful guarantee that goes beyond reporting "we saw a 5% gap in our test." It's more akin to how hardware or classical software is verified against specifications. The trade-off is that it requires many samples and is specific to the distribution tested, but it provides assurance that standard evaluations lack. The takeaway from certification work is that we can obtain quantitative guarantees on bias, at least under certain conditions, which is crucial for high-stakes deployments. Even if such methods are in early stages, they complement the picture by addressing the "worst-case" or probabilistic edge of bias: not just what average bias we observed, but what the maximum bias could be given what we have not observed.

In summary, the field now has a layered understanding of bias in LLMs (see Figure 7). At the representation level, biases are present and measurable in embeddings and model probabilities. At the output level, those biases do translate into harmful content or performance disparities in many scenarios. Large-scale evaluations confirm these issues are widespread but also show relative improvements as models are refined. And new methods like certification offer pathways to stronger assurances. This synthesis aligns with recent comprehensive surveys (Ferrara, 2023; Gallegos et al., 2024), which converge on the view that multiple methods must be used in concert to thoroughly evaluate bias. Intrinsic tests are fast and proactive; extrinsic tests are realistic and impact-oriented; and certification or stress-testing techniques add reliability guarantees. Together, they form a toolkit that is increasingly robust in characterizing where an LLM stands in terms of fairness.

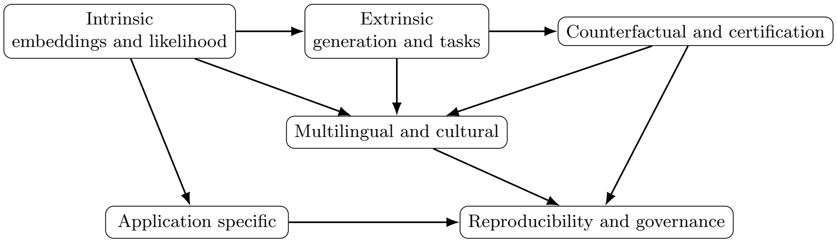

Intrinsic
embeddings and likelihood
Extrinsic
generation and tasks
Counterfactual and certification
Multilingual and cultural
Application specific
Reproducibility and governance

Figure 7. Method integration map. Intrinsic tests triage representational risks, extrinsic tests surface user-facing harms, and counterfactual or certification methods add statistical assurance. Multilingual, domain, and governance layers augment and stabilize the evaluation pipeline.

9.2 Distinguishing Model Bias from Societal Bias

An important conceptual question concerns whether observed biases in LLM outputs reflect the model's own distortion or merely mirror biases already present in the underlying population data (Bender et al., 2021; Blodgett et al., 2020; Mehrabi et al., 2021). In some cases, an LLM may generate statistically accurate but socially undesirable patterns because the training corpus itself encodes historical inequities and prejudiced discourse (Blodgett et al., 2020; Suresh & Guttag, 2021). In other cases, the model may amplify or distort those patterns beyond what is observed in real-world distributions, a phenomenon documented as bias amplification in prior work (Mehrabi et al., 2021; Zhao et al., 2017a). Distinguishing these scenarios is crucial for interpretation and for determining appropriate mitigation strategies.

From a measurement perspective, one approach is to compare model outputs against empirical population baselines or corpus-level statistics. For example, if occupational gender distributions in model outputs deviate substantially from real-world labor statistics, this may indicate amplification rather than simple reflection (Gallegos et al., 2024; Zhao et al., 2017a). Similarly, if the model produces disproportionately negative sentiment toward certain groups relative to corpus frequency or documented societal attitudes, this suggests an added model-level bias rather than faithful representation (Blodgett et al., 2020; Suresh & Guttag, 2021). Such baseline comparisons help separate descriptive alignment from normative distortion.

However, even faithful reflection of societal bias does not automatically absolve the model from responsibility. LLMs are not passive mirrors; they are deployed systems that shape user perceptions, decisions, and allocational outcomes (Barocas & Selbst, 2016; Suresh & Guttag, 2021). Therefore, evaluation frameworks must clarify whether fairness is defined relative to empirical reality, normative ideals, or regulatory standards. This distinction affects how bias metrics are interpreted and what counts as mitigation success (Blodgett et al., 2020; Gallegos et al., 2024).

In practice, bias audits should explicitly state whether they evaluate deviation from population statistics, amplification of harmful associations, or normative fairness criteria. Making this distinction transparent helps avoid conflating societal bias with model-induced bias and supports clearer communication with policymakers and stakeholders (Barocas & Selbst, 2016; Suresh & Guttag, 2021).

9.3 What Remains Hard

Despite significant progress, several challenges continue to vex researchers and practitioners in bias evaluation. Here we outline some of the persistent open problems and why they are difficult.

One fundamental issue is evaluator bias and construct validity—essentially, how do we ensure that our measurements of bias are themselves unbiased and truly reflective of harm? As discussed in Section 8, if we use an AI judge or a particular dataset as the gold standard, we might inadvertently be measuring the biases of those instruments rather than the model's bias. For instance, a toxicity detector might be more sensitive to profanity and thus flag outputs from certain groups as "toxic" more often, even if the content is not actually hateful. This could falsely make a model seem biased against that group. Ensuring validity often requires triangulation—using multiple indicators and involving human judgment to confirm whether what we label as "biased output" is genuinely problematic in context. However, this human involvement reintroduces subjectivity. In effect, we face the evaluative bias loop. No fully objective oracle for bias exists, because defining "bias" involves human values and norms. This ties into a larger point made by many (Blodgett et al., 2020): bias is inherently a social and contextual concept, so our evaluations will always have some normative assumptions. Developing evaluators that are as fair and context-aware as possible, perhaps via diverse human panels or improved AI judges, remains an open challenge.

Another hard problem is multilingual and cross-cultural measurement, which we detailed in Section 7. While we have extended evaluations to some languages, the coverage is very uneven. Many low-resource languages lack any bias benchmarks or even basic sentiment/toxicity lexicons. Additionally, societal biases differ—an expression that is considered a slur in one culture might not have an analogue in another. Evaluating an LLM's fairness in, e.g., Hindi or Swahili requires cultural competence and likely new methods. Automatic translation of test cases, although common, can fail because it does not capture nuance or because the model's performance in translation might mask its true behavior, e.g., the model can be very biased in Swahili, but when we translate its Swahili outputs to English, the translator masks the bias. There is also the issue of metrics: should we expect identical behavior across languages, e.g., equally low toxicity in English and Arabic, or should evaluations account for different baselines of training data, e.g., perhaps a model simply knows less about a rarer language, leading to different kinds of errors that complicate the bias picture? These questions do not have clear answers yet. What is clear is that multilingual fairness is far from solved: few LLMs have been rigorously audited in non-European languages, and early glimpses like biases in dialect as per Hofmann et al. 2024 suggest that significant issues lurk under the surface.

Third, intersectional and fine-grained group biases remain difficult to assess comprehensively. While we can run tests on many combinations, as the combinations grow, the data requirements explode and statistical power drops. Moreover, some intersections are hard to operate in prompts, e.g., how do we prompt for a combination of three or four attributes naturally? There's also the challenge of ethical and privacy considerations: explicitly testing sensitive combinations, e.g., religion plus sexual orientation, might produce content that is itself sensitive or offensive. Yet, if we avoid testing these, we might miss crucial failure modes. The field acknowledges intersectionality as important, but practical methodologies for robust intersectional audits are still being refined. This is an area where domain knowledge and community input are valuable—knowing which intersections are most salient can improve assessment. The theoretical difficulty is akin to the "fairness gerrymandering" problem (Kearns, Neel, Roth, & Wu, 2018), which showed that ensuring fairness on all individual attributes can still leave combined subgroup unfairness. In LLM terms, a model tuned to not be biased on single axes might still be biased on joint axes. Techniques to detect and mitigate that are still emerging.

One subtle open problem is to distinguish genuine fairness improvements from over-correction or reduced utility. As developers work to debias models, one worry is that they might achieve "fairness" by simply making the model very conservative or evasive whenever a sensitive topic arises. For example, early versions of ChatGPT would sometimes refuse to answer any question that mentioned a protected attribute. Superficially, this avoids producing a biased remark, but it introduces a new bias: differential treatment by selectively declining requests about certain groups or topics. If a model refuses to generate a story about two men getting married but is happy to do so for a man and woman, it's exhibiting a form of bias via differential refusal. However, if we only measure overt toxicity, we might falsely conclude the model is safe since it never produces toxic output about gay couples—it simply refuses to talk about them at all. This phenomenon—let's call it content suppression bias—is tricky to capture in evaluations. It requires metrics for when the model refuses or gives generic safe responses, and whether those occurrences correlate with certain groups. Some recent evaluations have started to include "refusal rate" or "hallucinated neutrality" as metrics. For instance, an evaluation might prompt the model: "Tell a joke about [group]" and see if the model disproportionately refuses for some groups out of caution. Balancing mitigation to avoid both harmful commission, e.g., saying something bad, and harmful omission, e.g., withholding or degrading service, is a nuanced challenge for LLM developers. As of now, few benchmarks systematically measure over-refusal or false compliance differences, so this remains an area for improvement. We highlight this because a model could appear unbiased under traditional tests but still be unfair by being overly restrictive in specific contexts—a kind of bias that standard metrics can easily miss.

Finally, there are theoretical limits and trade-offs that continue to loom over fairness in AI. The "impossibility results" in algorithmic fairness show that certain intuitive fairness criteria cannot all be satisfied simultaneously (Kleinberg, Mullainathan, & Raghavan, 2016). In the realm of LLMs, Anthis et al. (2024) argue that given the complexity of language and the numerous dimensions of potential bias, it may be fundamentally impossible for a single model to be entirely free of bias for all groups and contexts simultaneously. There will always be trade-offs—for example, making a model less biased in toxicity might inadvertently make it more biased in which questions it chooses to answer (the over-refusal problem). Another trade-off arises between specificity and generality: if you fine-tune a model to be fair on a particular benchmark, you might be narrowing its behavior in a way that could hurt performance or create other biases, like losing nuance in its responses. There's also the open question of to what extent language models can be fair if the underlying data (human language use) is biased. Some have posed that unless we fundamentally change training data or model architectures, we are always going to be post-hoc patching biases, a bit like a whack-a-mole game. These deep challenges do not have straightforward solutions. They remind us to be humble about what bias evaluations can achieve—they can show progress, but not perfection.

In sum, the difficult problems include ensuring our evaluations measure the "right" thing without injecting new bias; extending fairness across languages and cultures; capturing the full intersectional complexity of bias; avoiding Pyrrhic victories where reducing one bias introduces another form of harm; and grappling with inherent trade-offs that may make absolute fairness unattainable. These are active research frontiers. They suggest that bias evaluation will remain a dynamic field, needing continual refinement and perhaps new paradigms, e.g., more human-AI collaborative evaluation, or periodic reevaluation as societal norms evolve. Recognizing these challenges is important for practitioners so they approach bias mitigation with caution and awareness that an “all clear” on current metrics does not guarantee the absence of problems. Figure 8 below proposes an incremental program for bias evaluation maturity. One can locate their current phase and identify next steps, e.g., moving from broad screening to certification for high-stakes deployments.

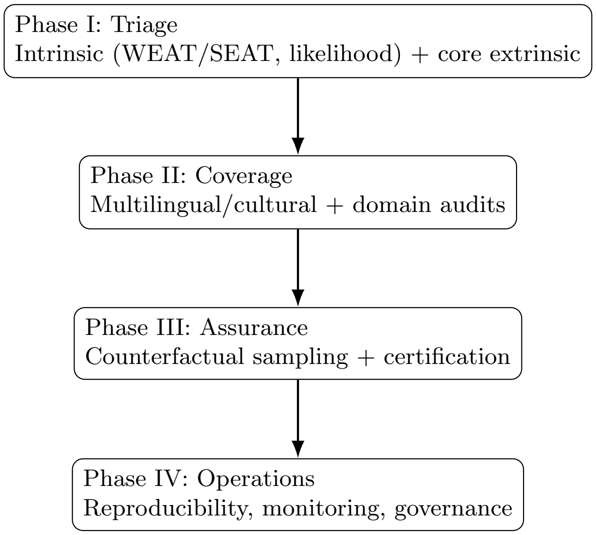

Phase I: Triage
Intrinsic (WEAT/SEAT, likelihood) + core extrinsic
Phase II: Coverage
Multilingual/cultural + domain audits
Phase III: Assurance
Counterfactual sampling + certification
Phase IV: Operations
Reproducibility, monitoring, governance

Figure 8. Roadmap from triage to assurance and operations. Each phase builds on the last: start with broad screening, increase coverage, add guarantees for critical specs, and institutionalize reproducibility and governance.

9.4 Practitioner Checklist

In light of our comprehensive review, we distill here a practical checklist for practitioners who wish to evaluate and mitigate bias in LLMs. This checklist is a set of concrete recommendations, synthesizing the insights from all sections into actionable guidance.

– *Use multiple evaluation methods in tandem.* Do not rely on a single metric or dataset to assess bias. Combine intrinsic tests, e.g., embedding association metrics and likelihood-based bias scores with extrinsic evaluations, e.g., prompt-based generation tests, task performance gaps. For example, run WEAT or SEAT to probe embeddings and also test with a stereotyping benchmark like CrowS-Pairs or BBQ. Consistent findings across methods greatly strengthen conclusions, while divergences can reveal nuances (Sections 4 and 5).

- *Prioritize counterfactual paired testing for salient biases.* Wherever possible, structure your evaluation around paired examples that differ only in a sensitive attribute. This could be as simple as comparing model outputs for "he" vs "she" in a template, or as complex as generating matched profiles for candidates of different races in a hiring scenario. Paired tests directly measure bias as a difference in output, making interpretation more straightforward (Section 6). If you have limited resources, focus on a few high-impact bias scenarios and create counterfactual pairs for them – this often provides clear evidence of any disparity.

- *Include uncertainty and significance in reporting results.* Always accompany bias metrics with confidence intervals or statistical tests. Instead of stating "Model X is less toxic for group A than B," say "Model X showed a 4% (±2%) lower toxicity rate for group A vs. B in our sample." This communicates the reliability of the measurement. If results are not statistically significant, treat them with caution and possibly gather more data. Attaching uncertainty is especially important for small subgroup evaluations and for new models where variance might be high (Section 5).

- *Leverage bias certificates for high-stakes deployments.* If you are working with an application where fairness is mission-critical, e.g., an AI system used in hiring, lending, or healthcare advice, consider using formal methods like LLMCert-B or extensive stress testing to obtain a bias guarantee. While these require more effort, they can provide assurances like "with 99% confidence, the model's predictions meet fairness criterion X." Even if you cannot do this for every bias aspect, doing it for the most critical one, e.g., gender fairness in loan recommendations, adds a layer of trust and is increasingly expected in regulated industries (Section 6).

- *Regularly audit and document bias evaluations as part of model development.* Do not treat bias testing as a one-off task. Incorporate it into model iteration cycles. Each time the model architecture is changed or it's fine-tuned on new data, re-run the suite of bias tests to catch regressions or new issues. Maintain a "bias evaluation card" (Section 8's reproducibility checklist) for the model, which logs when and how bias was evaluated and what changed over time. This not only helps internally but also fulfills transparency requirements for governance.

- *Align bias evaluation with governance frameworks and stakeholder values.* Choose evaluation targets and thresholds that make sense in the context of use and according to any ethical guidelines or laws you operate under. For instance, if deploying a chatbot in the EU, ensure your bias tests cover all EU protected characteristics, since the AI Act will expect that. Involve representatives from affected communities when designing or reviewing bias tests—they might point out biases or harms you did not consider initially. Ultimately, the goal is not just to "pass benchmarks" but to ensure the model is fair in the eyes of those who use or are impacted by it.

Table 6 maps common deployment contexts to concrete method bundles. It is intended as a quick-start guide for selecting an evaluation plan aligned with resources, risks, and regulatory expectations.

Table 6. Context-aware selection of bias evaluation methods.

Context	Recommended methods
Early model triage	Embedding and sentence association tests such as WEAT and SEAT, likelihood-based tests, a small sweep of counterfactual pairs, and basic generation toxicity or regard analysis with confidence intervals.
Multilingual deployment	Localized prompt sets, per-language calibration of bias and toxicity detectors, and stratified human validation across languages, dialects, and groups.
High-stakes domain	Task-grounded vignettes with parity checks for accuracy and decision gaps, targeted stress tests, and certification-style evaluation with specified metrics and input distributions.
Governance-ready release	A multi-metric report with uncertainty estimates, a model and bias card, released code and configuration artifacts, and a documented monitoring and escalation plan.

By following this checklist, practitioners can systematically evaluate bias and work towards mitigating it. The recommendations emphasize a proactive, rigorous, and context-aware approach—evaluating from multiple angles, quantifying confidence in findings, and iterating as needed. It is worth noting that bias evaluation is an ongoing responsibility: as LLMs are updated or encounter new real-world data, new biases can emerge, and societal norms of fairness may shift. Therefore, treating bias evaluation as a continuous process is the best practice.

Final Thoughts Bias in LLMs is a complex, multifaceted problem at the intersection of technology and society. Through this review, we have assembled a broad toolkit to detect and quantify biases, from internal representations to external behaviors, and then to certify model fairness properties. We have also identified the limitations of these methods and the challenges that lie ahead. For practitioners, the path forward involves using these tools in combination, remaining vigilant about new forms of bias, and engaging with the wider community–including policymakers and affected users–to define what fairness means for each application. By doing so, we move toward LLM deployments that are not only innovative, but also equitable and worthy of the trust of the society.

References

Abid, A., Farooqi, M., & Zou, J. (2021). Large language models associate muslims with violence. *Nature Machine Intelligence*, *3*(6), 461–463. doi: https://doi.org/10.1038/s42256-021-00356-9

An, J., Huang, D., Lin, C., & Tai, M. (2025, February). Measuring gender and racial biases in large language models: Intersectional evidence from automated resume evaluation. *PNAS Nexus*, *4*(3). Retrieved from http://dx.doi.org/10.1093/pnasnexus/pgaf089 doi: https://doi.org/10.1093/pnasnexus/pgaf089

Anthis, J., Lum, K., Ekstrand, M., Feller, A., D'Amour, A., & Tan, C. (2024). The impossibility of fair LLMs. *arXiv*. Retrieved from https://arxiv.org/abs/2406.03198 doi: https://doi.org/10.18653/v1/2025.acl-long.5

Barocas, S., & Selbst, A. D. (2016). Big data's disparate impact. *California Law Review*, *104*(3), 671–732. doi: https://doi.org/10.2139/ssrn.2477899

Bartl, M., Nissim, M., & Gatt, A. (2020). Unmasking contextual stereotypes: Measuring and mitigating bert's gender bias. In *Proceedings of the second workshop on gender bias in natural language processing* (pp. 1–16). Barcelona, Spain (Online): Association for Computational Linguistics. Retrieved from https://aclanthology.org/2020.gebnlp-1.1/

Bastani, O., Zhang, X., & Solar-Lezama, A. (2019). Probabilistic verification of fairness properties via concentration. *Proceedings of the ACM on Programming Languages*, *3*(OOPSLA), 118:1–118:27. Retrieved from https://dl.acm.org/doi/10.1145/3360544 doi: https://doi.org/10.1145/3360544

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 acm conference on fairness, accountability, and transparency (facct)* (pp. 610–623). doi: https://doi.org/10.1145/3442188.3445922

Blodgett, S. L., Barocas, S., Daumé III, H., & Wallach, H. (2020). Language (technology) is power: A critical survey of bias in NLP. In *Proceedings of the 58th annual meeting of the association for computational linguistics (acl)* (pp. 5454–5476). doi: https://doi.org/10.18653/v1/2020.acl-main.485

Bolukbasi, T., Chang, K., Zou, J., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems 29 (neurips 2016)* (pp. 4349–4357).

Bordia, S., & Bowman, S. R. (2019). Identifying and reducing gender bias in word-level language models. *arXiv*. Retrieved from https://arxiv.org/abs/1904.03035 doi: https://doi.org/10.18653/v1/n19-3002

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency* (pp. 77–91).

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, *356*(6334), 183–186. doi: https://doi.org/10.1126/science.aal4230

Cao, Y., Pruksachatkun, Y., Chang, K., Gupta, R., Kumar, V., Dhamala, J., & Galstyan, A. (2022). On the intrinsic and extrinsic fairness evaluation metrics for contextualized language representations. In *Proceedings of acl 2022 (short papers)* (pp. 561–570). doi: https://doi.org/10.18653/v1/2022.acl-short.62

Chaudhary, I., Hu, Q., Kumar, M., Ziyadi, M., Gupta, R., & Singh, G. (2025). Certifying counterfactual bias in LLMs. In *International conference on learning representations (iclr).* (OpenReview)

Crenshaw, K. (1991, July). Mapping the margins: Intersectionality, identity politics, and violence against women of color. *Stanford Law Review*, *43*(6), 1241–1299. Retrieved from `http://dx.doi.org/10.2307/1229039` doi: https://doi.org/10.2307/1229039

Cui, J., Chiang, W.-L., Stoica, I., & Hsieh, C.-J. (2025). OR-Bench: An over-refusal benchmark for large language models. In *Proceedings of the 42nd international conference on machine learning (icml).* (arXiv:2405.20947)

De-Arteaga, M., Romanov, A., Wallach, H., Chayes, J., Borgs, C., Kalai, A., & Crawford, K. (2019). Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the acm conference on fairness, accountability, and transparency (fat*).* doi: https://doi.org/10.1145/3287560.3287572

Dev, S., & Phillips, J. M. (2019). Attenuating bias in word vectors. In *Proceedings of the 22nd international conference on artificial intelligence and statistics (aistats)* (pp. 879–887).

Dhamala, J., Sun, T., Kumar, V., Krishna, S., Pruksachatkun, Y., Chang, K., & Gupta, R. (2021). BOLD: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 acm conference on fairness, accountability, and transparency (facct).* doi: https://doi.org/10.1145/3442188.3445924

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference (itcs)* (pp. 214–226). (Preprint 2011) doi: https://doi.org/10.1145/2090236.2090255

Ethayarajh, K. (2019). How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 55–65). Hong Kong, China: Association for Computational Linguistics. Retrieved from `https://aclanthology.org/D19-1006/` doi: https://doi.org/10.18653/v1/D19-1006

European Parliament, & Council of the European Union. (2024). Regulation (EU) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence and amending regulations (EC) no 300/2008, (EU) 2017/745, (EU) 2017/746, (EU) 2019/881 and (EU) 2022/2065 and directive 2009/125/ec (Artificial Intelligence Act). *Official Journal of the European Union, L 2024/1689*. Retrieved from https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng

Ferrara, E. (2023). Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies. *arXiv*. Retrieved from https://arxiv.org/abs/2304.07683 doi: https://doi.org/10.3390/sci6010003

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M. M., Kim, S., Dernoncourt, F., ... Ahmed, N. K. (2024). Bias and fairness in large language models: A survey. *Computational Linguistics*, *50*(3), 1097–1158. doi: https://doi.org/10.1162/coli_a_00524

Gehman, S., Gururangan, S., Sap, M., Choi, Y., & Smith, N. A. (2020). Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the association for computational linguistics: Emnlp 2020* (pp. 3356–3369). doi: https://doi.org/10.18653/v1/2020.findings-emnlp.301

Gumilar, K. E., Indraprasta, B. R., Hsu, Y.-C., Yu, Z.-Y., Chen, H., Irawan, B., ... Tan, M. (2024, July). Disparities in medical recommendations from AI-based chatbots across different countries/regions. *Scientific Reports*, *14*(1). Retrieved from http://dx.doi.org/10.1038/s41598-024-67689-0 doi: https://doi.org/10.1038/s41598-024-67689-0

Guo, Y., Guo, M., Su, J., Yang, Z., Zhu, M., Li, H., & Qiu, M. (2024). Bias in large language models: Origin, evaluation, and mitigation. *arXiv*. Retrieved from https://arxiv.org/abs/2411.10915

Hanu, L., & Unitary team. (2020). *Detoxify.* https://github.com/unitaryai/detoxify. Retrieved from https://github.com/unitaryai/detoxify (GitHub repository)

Hofmann, V., Kalluri, P. R., Jurafsky, D., & King, S. (2024). Dialect prejudice predicts AI decisions about people's character, employability, and criminality. In *Proceedings of the 2024 acm conference on fairness, accountability, and transparency (facct)* (pp. 1321–1340).

Huang, P., Zhang, H., Jiang, R., Stanforth, R., Welbl, J., Rae, J., ... Kohli, P. (2019). Reducing sentiment bias in language models via counterfactual evaluation. *arXiv*. Retrieved from https://arxiv.org/abs/1911.03064 doi: https://doi.org/10.18653/v1/2020.findings-emnlp.7

Kearns, M., Neel, S., Roth, A., & Wu, Z. S. (2018, Jul). Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th international conference on machine learning* (Vol. 80, pp. 2564–2572). PMLR. Retrieved from https://proceedings.mlr.press/v80/kearns18a.html

Kleinberg, J., Mullainathan, S., & Raghavan, M. (2016). *Inherent trade-offs in the fair determination of risk scores.* Retrieved from https://arxiv.org/abs/1609.05807

Kotek, H., Dockum, R., & Sun, D. (2023). Gender bias and stereotypes in large language models. In *Proceedings of the acm collective intelligence conference (ci 2023).* doi: https://doi.org/10.1145/3582269.3615599

Krishna, S., Gupta, R., Verma, A., Dhamala, J., Pruksachatkun, Y., & Chang, K.-W. (2022, may). Measuring fairness of text classifiers via prediction sensitivity. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 5830–5842). Dublin, Ireland: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2022.acl-long.401/ doi: https://doi.org/10.18653/v1/2022.acl-long.401

Kurita, K., Vyas, N., Pareek, A., Black, A. W., & Tsvetkov, Y. (2019). Measuring bias in contextualized word representations. In *Proceedings of the first acl workshop on gender bias for nlp* (pp. 166–172). doi: https://doi.org/10.18653/v1/w19-3823

Kusner, M. J., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual fairness. In *Advances in neural information processing systems 30 (neurips 2017)* (pp. 4066–4076).

Lee, G., Hartmann, V., Park, J., Papailiopoulos, D., & Lee, K. (2023). Prompted llms as chatbot modules for long open-domain conversation. In *Findings of the association for computational linguistics: Acl 2023* (pp. 4536–4554). Association for Computational Linguistics. Retrieved from https://aclanthology.org/2023.findings-acl.277/ doi: https://doi.org/10.18653/v1/2023.findings-acl.277

Li, Y., Du, M., Song, R., Wang, X., & Wang, Y. (2024). A survey on fairness in large language models. *arXiv*. Retrieved from https://arxiv.org/abs/2308.10149 (Version 2 (revised 2024-02-21)) doi: https://doi.org/10.48550/arXiv.2308.10149

Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., . . . Koreeda, Y. (2023). Holistic evaluation of language models. *Transactions on Machine Learning Research*. Retrieved from https://arxiv.org/abs/2211.09110 (TMLR; arXiv:2211.09110) doi: https://doi.org/10.48550/arXiv.2211.09110

Liu, T., Luo, R., Chen, Q., Qin, Z., Sun, R., Yu, Y., & Zhang, C. (2024). Jailbreaking black-box large language models in twenty queries. In *33rd usenix security symposium (usenix security 24)*. Philadelphia, PA: USENIX Association. Retrieved from https://www.usenix.org/conference/usenixsecurity24/presentation/liu-tong (See also arXiv:2310.08419)

Liu, Y., Yang, T., Huang, S., Zhang, Z., Huang, H., Wei, F., . . . Zhang, Q. (2023). *Calibrating llm-based evaluator.* Retrieved from https://arxiv.org/abs/2309.13308

May, C., Wang, A., Bordia, S., Bowman, S. R., & Rudinger, R. (2019). On measuring social biases in sentence encoders. In *Proceedings of naacl-hlt 2019* (pp. 622–628). doi: https://doi.org/10.18653/v1/n19-1063

Meade, N., Poole-Dayan, E., & Reddy, S. (2022). An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1878–1898). Dublin, Ireland: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2022.acl-long.132/ (Published at ACL 2022; widely cited in 2023 literature) doi: https://doi.org/10.18653/v1/2022.acl-long.132

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys*, *54*(6), 1–35. doi: https://doi.org/10.1145/3457607

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., . . . Gebru, T. (2019). Model cards for model reporting. In *Proceedings of the 2019 conference on fairness, accountability, and transparency (fat*)* (pp. 220–229). doi: https://doi.org/10.1145/3287560.3287596

Nadeem, M., Bethke, A., & Reddy, S. (2021). Stereoset: Measuring stereotypical bias in pretrained language models. In *Proceedings of acl 2021 (long papers)* (pp. 5356–5371). doi: https://doi.org/10.18653/v1/2021.acl-long.416

Nangia, N., Vania, C., Bhalerao, R., & Bowman, S. R. (2020). Crows-pairs: A challenge dataset for measuring social biases in masked language models. In *Findings of the association for computational linguistics: Emnlp 2020* (pp. 227–239). doi: https://doi.org/10.18653/v1/2020.emnlp-main.154

Panickssery, A., Bowman, S. R., & Feng, S. (2024). *Llm evaluators recognize and favor their own generations.* Retrieved from https://arxiv.org/abs/2404.13076 doi: https://doi.org/10.52202/079017-2197

Parrish, A., Chen, A., Nangia, N., Padmakumar, V., Phang, J., Thompson, J., . . . Bowman, S. (2022). BBQ: A hand-built bias benchmark for question answering. In *Findings of the association for computational linguistics: Acl 2022* (pp. 2086–2105). doi: https://doi.org/10.18653/v1/2022.findings-acl.165

Perez, E., Huang, S., Song, F., Cai, T., Ring, R., Aslanides, J., . . . Irving, G. (2022). *Red teaming language models with language models.* Retrieved from https://arxiv.org/abs/2202.03286 doi: https://doi.org/10.18653/v1/2022.emnlp-main.225

Raji, I. D., Denton, E., Bender, E. M., Hanna, A., & Paullada, A. (2021). AI and the everything in the whole wide world benchmark: A critical analysis of the biggest benchmarks in AI. *arXiv*. Retrieved from https://arxiv.org/abs/2111.15366 (Metric validity discussion)

Ravfogel, S., Elazar, Y., Gonen, H., Twiton, M., & Goldberg, Y. (2020). Null it out: Debiasing text representations by iterative nullspace projection. In *Proceedings of the 58th annual meeting of the association for computational linguistics (acl)* (pp. 7237–7256).

Rozado, D. (2025). *Gender and positional biases in llm-based hiring decisions: Evidence from comparative cv/résumé evaluations.* Retrieved from https://arxiv.org/abs/2505.17049 doi: https://doi.org/10.7717/peerj-cs.3628

Rudinger, R., Naradowsky, J., Leonard, B., & Van Durme, B. (2018). Gender bias in coreference resolution. In *Proceedings of the 2018 conference of the north american chapter of the association for computational* *linguistics: Human language technologies, volume 2 (short papers)* (pp. 8–14). New Orleans, Louisiana: Association for Computational Linguistics. Retrieved from https://aclanthology.org/N18-2002/ doi: https://doi.org/10.18653/v1/N18-2002

Rupprecht, J., Ahnert, G., & Strohmaier, M. (2025). *Prompt perturbations reveal human-like biases in large language model survey responses.* Retrieved from https://arxiv.org/abs/2507.07188

Sheng, E., Chang, K., Natarajan, P., & Peng, N. (2019). The woman worked as a babysitter: On biases in language generation. In *Proceedings of emnlp-ijcnlp 2019* (pp. 3407–3412). doi: https://doi.org/10.18653/v1/d19-1339

Sherry, J. H. (1965). The civil rights act of 1964: Fair employment practices under title VII. *Cornell Hotel and Restaurant Administration Quarterly*, *6*(2), 3–6. doi: https://doi.org/10.1177/001088046500600202

Sim, J., & Reid, N. (1999). Statistical inference by confidence intervals: Issues of interpretation and utilization. *Physical Therapy*, *79*(2), 186–195. doi: https://doi.org/10.1093/ptj/79.2.186

Smith, E. M., Hall, M., Kambadur, M., Presani, E., & Williams, A. (2022). I'm sorry to hear that: Finding new biases in language models with a holistic descriptor dataset. *arXiv*. Retrieved from https://arxiv.org/abs/2201.11745 doi: https://doi.org/10.18653/v1/2022.emnlp-main.625

Solaiman, I., Brundage, M., Clark, J., Askell, A., Herbert-Voss, A., Wu, J., ... Wang, J. (2019). Release strategies and the social impacts of language models. *arXiv*. Retrieved from https://arxiv.org/abs/1908.09203

Suresh, H., & Guttag, J. V. (2021). A framework for understanding sources of harm throughout the machine learning life cycle. In *Proceedings of acm eaamo 2021.* (Article 7) doi: https://doi.org/10.1145/3465416.3483305

Tabassi, E. (2023). *Artificial intelligence risk management framework (ai rmf 1.0).* Retrieved from http://dx.doi.org/10.6028/NIST.AI.100-1 doi: https://doi.org/10.6028/nist.ai.100-1

Vig, J., Gehrmann, S., Belinkov, Y., Qian, S., Nevo, D., Singer, Y., & Shieber, S. (2020). Investigating gender bias in language models using causal mediation analysis. In *Advances in neural information processing systems 33 (neurips 2020).*

Wang, B., Chen, W., Pei, H., Xie, C., Kang, M., Zhang, C., ... Li, B. (2024). Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv*. Retrieved from https://arxiv.org/abs/2306.11698

Zhang, Y., Huang, Y., Sun, Y., Liu, C., Zhao, Z., Fang, Z., ... Zhu, J. (2024). Multitrust: A comprehensive benchmark towards trustworthy multimodal large language models. *arXiv*. Retrieved from https://arxiv.org/abs/2406.07057 doi: https://doi.org/10.52202/079017-1561

Zhao, J., Wang, T., Yatskar, M., Cotterell, R., Ordonez, V., & Chang, K. (2019). Gender bias in contextualized word embeddings. *arXiv*. Retrieved from https://arxiv.org/abs/1904.03310 doi: https://doi.org/10.18653/v1/n19-1064

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K. (2017a). Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 conference on empirical methods in natural language processing (emnlp).*

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K. (2018). Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of naacl-hlt 2018* (pp. 15–20). doi: https://doi.org/10.18653/v1/n18-2003

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K.-W. (2017b). Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2979–2989). Association for Computational Linguistics. Retrieved from http://dx.doi.org/10.18653/v1/D17-1323 doi: https://doi.org/10.18653/v1/d17-1323

Zollo, T. P., Morrill, T., Deng, Z., Snell, J. C., Pitassi, T., & Zemel, R. (2024). Prompt risk control: A rigorous framework for responsible deployment of large language models. *arXiv*. Retrieved from https://arxiv.org/abs/2311.13628