

# An End-To-End, Stage-Wise Life-Cycle Framework for Mitigating Algorithmic Bias in Two-Sided Marketplaces

**Nihar Vipulkumar Patel**

## **Abstract**

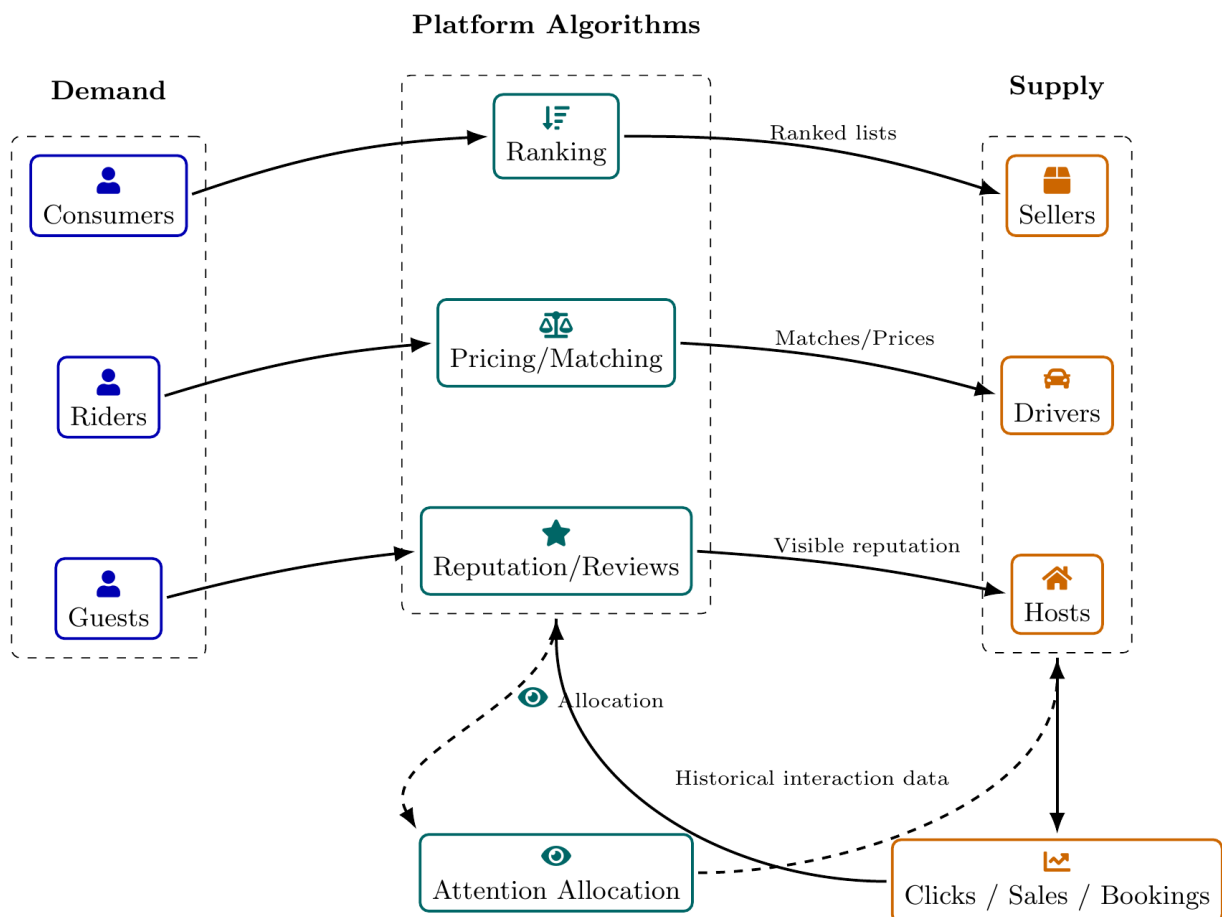
Two-sided marketplaces connect buyers and sellers by deciding who sees what, how things are priced, and how feedback affects future visibility. Those feedback loops can snowball: small early advantages like belonging to a well-represented group lead to more views, faster ratings, and better liquidity, which in turn win even more exposure. Meanwhile, newcomers or minority groups can lag despite offering similar quality. Over time, these outcomes can feel unfair, chip away at user trust, and harm the platform's long-term health. This paper lays out a practical, end-to-end approach to reduce algorithmic bias across the marketplace's entire lifecycle from how data is collected to how the system is monitored and governed. It treats "exposure" (who gets seen) and "opportunity" (who gets a real shot) as core resources, and sets goals that account for different user segments and calibration needs. The approach maps concrete actions across eight stages: measuring disparities; curating datasets; learning fair representations; training with constraints; ranking and allocating with fairness-aware exploration; shaping feedback signals; evaluating with counterfactual methods; and governing after launch. Crucially, bias is framed as a system-level, dynamic effect. It is not merely a flaw in a single model. Fairness constraints are applied to exposure flows, not only to predictions. Counterfactual simulations estimate long-term marketplace effects, and "pacers" limit how fast policies change so the user experience stays stable. The paper also shows how to connect offline guarantees with online tests, using techniques like propensity-scored log replay, interleaving experiments with exposure budgets, and drift alerts for segment-level calibration. The outcome is an architecture that spreads responsibility across teams and components, helping platforms reduce real disparities and perceived unfairness without tanking efficiency. It closes with guidance on choosing fairness trade-offs, keeping operational runbooks current, and defining leading indicators that tie short-term metrics to long-term trust.

**Key Words:** Algorithmic fairness, Two-sided marketplaces, Exposure bias, Dynamic feedback loops, Fairness-aware ranking, Counterfactual evaluation, Trust and governance

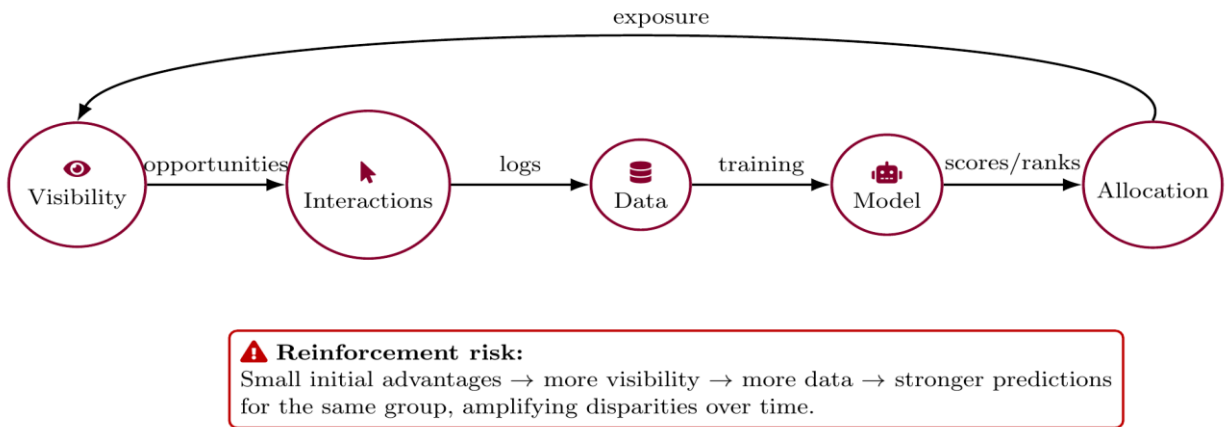
## 1. Introduction

Two-sided online marketplaces are a category of online platforms that facilitate exchange between supply and demand parties, including ridesharing platforms connecting riders and drivers, e-commerce sites connecting buyers and sellers, and rental services connecting guests and hosts. These systems often rely on algorithmic systems to mediate discovery between supply and demand and typically consist of ranking algorithms, pricing or matching mechanisms, and reputation or review systems. Algorithms learn from historical interactions in which certain suppliers and consumers found matches and optimize future recommendations or matches to improve overall platform outcomes, such as higher conversion rates or revenue. Because the algorithms determine who and what is visible to users, they effectively allocate a scarce resource: attention. This allocation process creates a feedback loop wherein increased visibility leads to more opportunities for success (e.g., clicks, sales, bookings), which in turn generates additional data that the algorithm uses to refine or reinforce its predictions. Over time, small initial differences between marketplace participants can become amplified through this loop.

One key issue is that these feedback loops may create unfair advantage for certain groups of participants over others. As a hypothetical example, imagine that providers from the majority group or those from regions with large numbers of early adopters get excessive initial exposure, build up more reviews and higher reputations, and continue to attract a disproportionate share of consumer attention. In contrast, providers from minority or historically disadvantaged groups, or new entrants without much history, may receive low initial exposure and less opportunity to demonstrate their quality, leading to persistent worse outcomes. These inequities can occur even in cases where the algorithms running the platform are not explicitly biased against any particular group (Koutroumpis, Leiponen, and Thomas 2020). Instead, they are a product of the learning dynamics and the predictable nature of prior data. In two-sided marketplaces, algorithmic bias usually arises from reinforcement dynamics and data feedback loops rather than overt discrimination coded into the system itself.



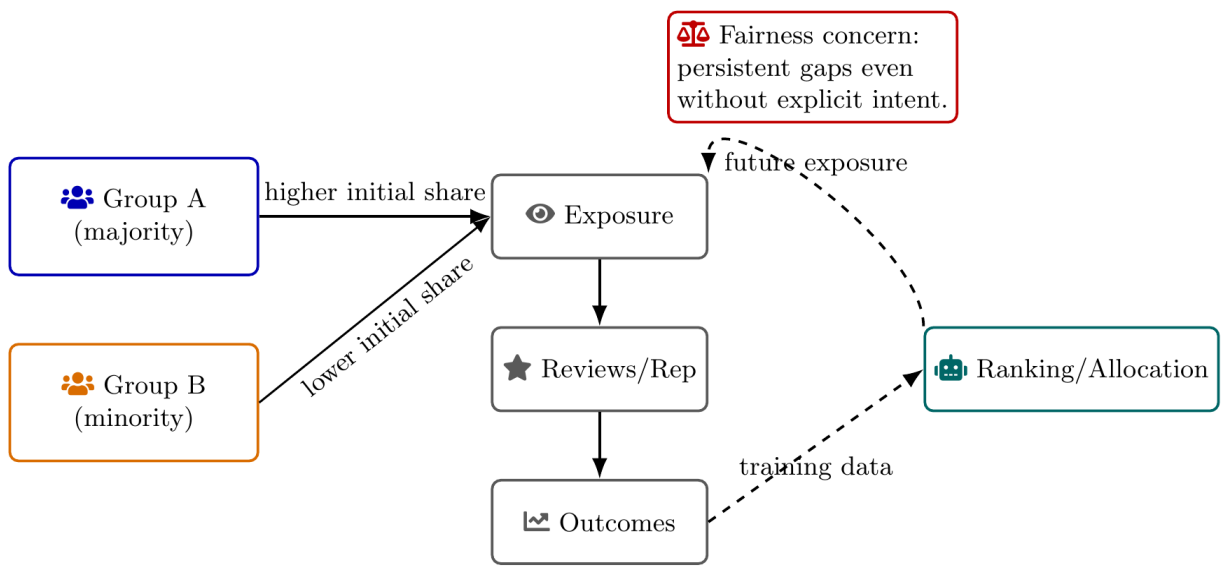
**Figure 1:** Two-sided marketplace mediation between demand and supply via ranking, pricing/matching, and reputation. Aggregated feedback paths reduce clutter.



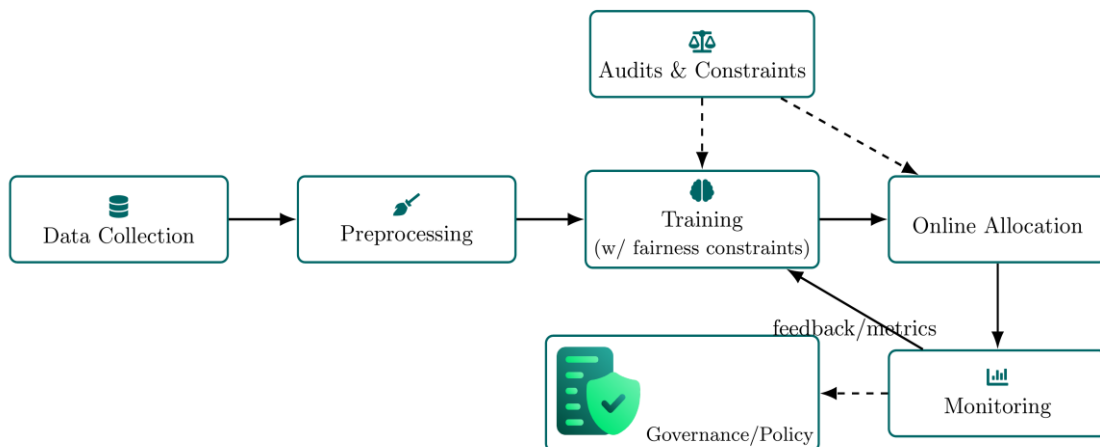
**Figure 2:** Reinforcement loop: visibility drives interactions and data, which train the model to allocate future exposure, closing the loop and potentially amplifying initial differences.

Bias, in this context, is not just an issue of statistical error or model fit; it reflects an amplified inequity in economic opportunity on the platform. Left unchecked, such bias threatens to undermine perceived fairness in the marketplace. Sellers or providers who may feel that they "never get a chance" because the algorithm rarely surfaces their offerings will lose faith in the platform and ultimately exit, reducing supply diversity. Buyers or consumers may similarly view the marketplace as narrow or unfair when they encounter the same dominant providers or a lack of diversity over and over again in the available content. Over time, these perceptions erode the user base and threaten to undermine the reputation and sustainability of the platform.

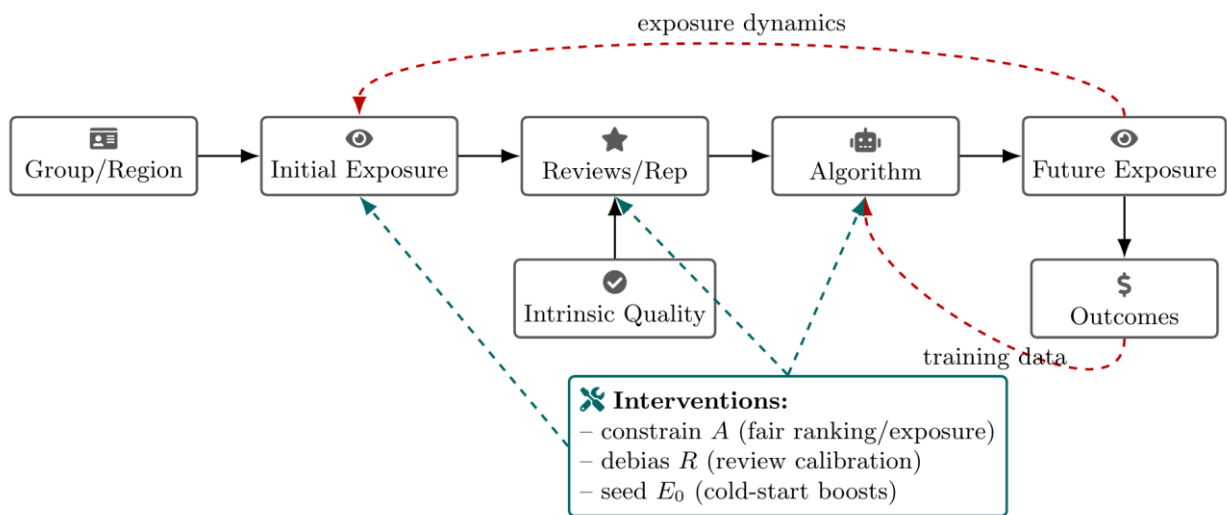
Addressing algorithmic bias in marketplaces is complex because interventions at any single point—for instance, adapting the ranking algorithm—may not be enough. The biases are systemic, emerging from interdependent components comprising data collection processes, model training procedures, real-time allocation mechanisms, user feedback systems, and even policy and governance decisions. For this reason, a holistic lifecycle approach is needed—one that accounts for bias mitigation at every step of the development and operation of the system. Such an approach requires reconsideration of how data are collected and preprocessed, how models are trained using fairness constraints, how live systems allocate exposure and opportunities, and how outcomes are monitored and governed over time. This paper presents a holistic, stage-wise framework for bias mitigation in two-sided marketplaces across a product’s entire lifecycle. The problem is decomposed into eight stages of bias that may be introduced or amplified at a specific point in the system. Going beyond traditional single-model fairness adjustments, this framework views exposure (the opportunity for items or participants to be seen) as a resource that needs fair allocation and enforces fairness constraints down the pipeline. Bias is treated as a dynamic property of the marketplace ecosystem rather than as a static one, such as a single fairness metric applied to model predictions. Fairness in the marketplace is thus an evolving outcome of continuous interactions and feedback loops.



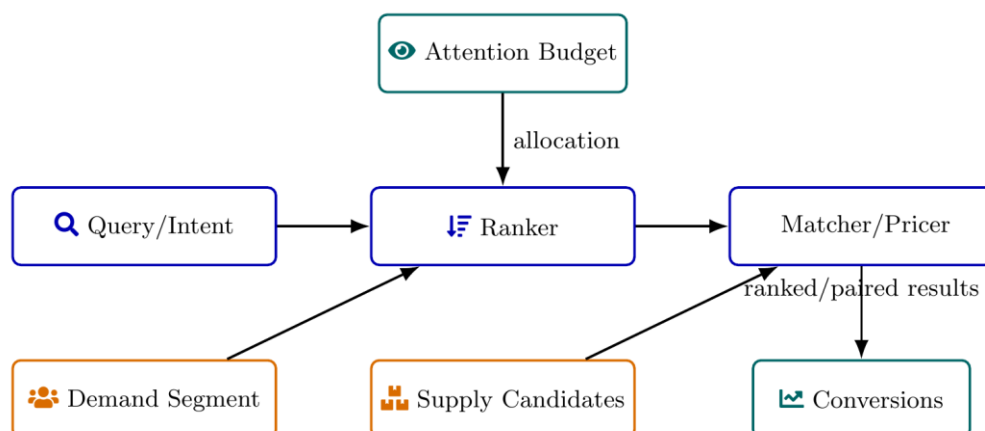
**Figure 3:** Disparity amplification: initial exposure gaps between groups can compound via reviews/reputation and outcomes, then feed back into the allocation algorithm.



**Figure 4:** Lifecycle approach to bias mitigation: address issues at collection, preprocessing, training (with constraints), online allocation, monitoring, and governance. This approach also includes recurring audits.



**Figure 5:** Structural view of marketplace dynamics: group membership and initial exposure influence reviews and the algorithm, which sets future exposure and outcomes; dashed edges show feedback and potential intervention points.



**Figure 6:** End-to-end discovery and allocation: queries and segments meet supply under a limited attention budget, producing conversions that feedback as training signals.

The rest of the paper is organized as follows. In Section 2, we will formalize the problem setting and the threat model, including ways in which bias can manifest and cascade via a two-sided marketplace. Then we describe the

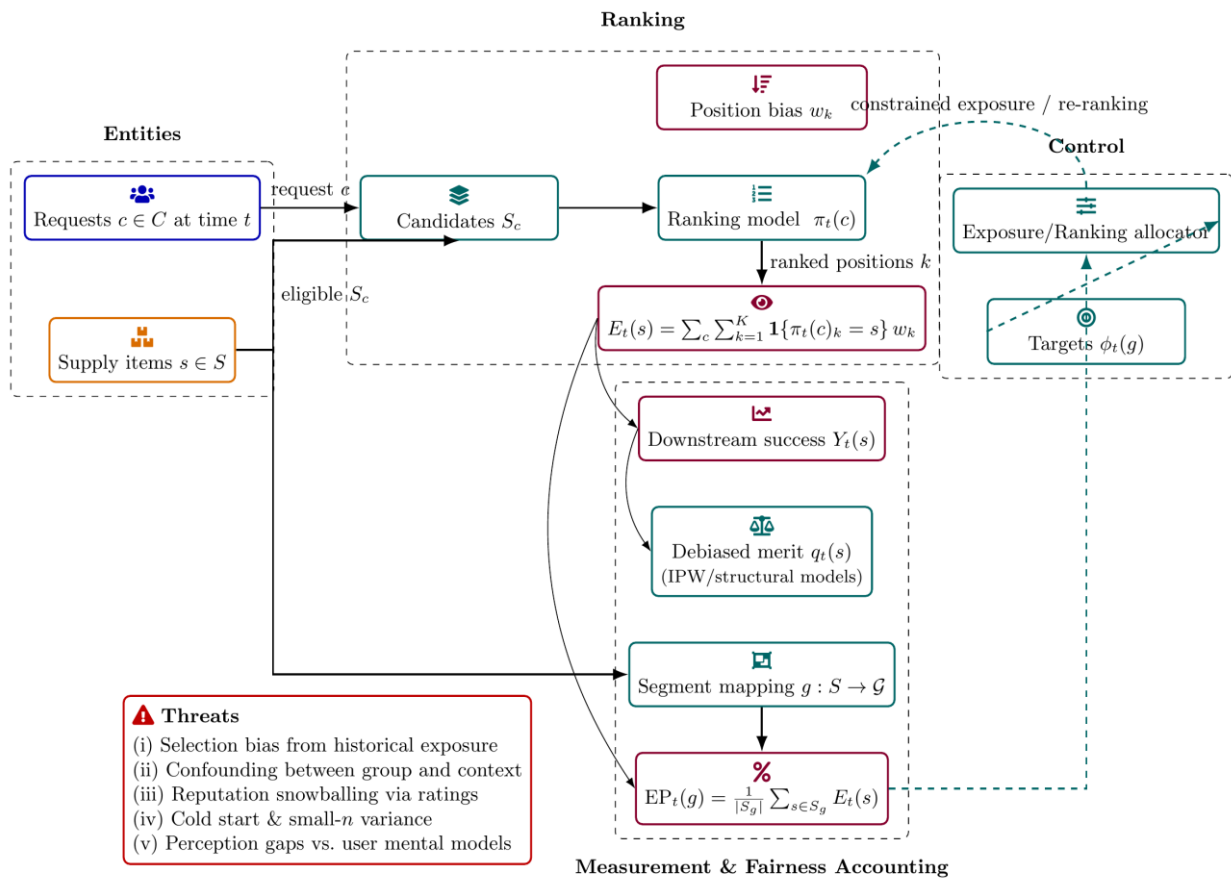
stage-by-stage framework and the eight stages of intervention across a full lifecycle of measurement and data preparation through model training, deployment, and governance in Section 3. In Sections 4 and 5, we present practical considerations for bridging offline fairness guarantees to online systems and structuring teams and responsibilities to operationalize the framework. Section 6 discusses the limitations and open challenges of the current work. Finally, Section 7 concludes with a summary and reflections on how these methods align short-term performance metrics with long-term goals of fairness and trust within marketplace ecosystems. (Reurink 2018)

## 2. Problem Formulation and Threat Model

We consider a two-sided marketplace with a set of supply items  $S$  (e.g. providers or listings) and a set of consumers  $C$  (e.g. user queries or requests). For each incoming consumer request  $c \in C$  at time  $t$ , the platform’s ranking or matching algorithm produces an ordered list (a permutation) of candidate items  $\pi_t(c) = (s_1, s_2, \dots, s_K)$  from the available supply  $S_c \subseteq S$  relevant to  $c$ . The item at rank  $k$  in this list is denoted  $\pi_t(c)_k = s_k$ . When an item  $s$  is presented to a consumer, it gains an opportunity for a conversion or success event (such as a click, purchase, or booking). We define the exposure for item  $s$  at time  $t$  as the cumulative attention it receives across all consumer requests at that time:

$$E_t(s) = \sum_{c \in C} \sum_{k=1}^K \mathbf{1}\{\pi_t(c)_k = s\} w_k,$$

where  $w_k$  is a position bias weight that captures the fact that being shown at rank  $k$  yields a certain fraction of a consumer’s attention (for example, items at the top of a list receive more attention than those at the bottom). In effect,  $E_t(s)$  aggregates the exposure of item  $s$  by summing over all impressions it gets, weighted by the prominence of its position in each ranking.



**Figure 7:** Pipeline for exposure fairness in a two-sided marketplace. A ranking model outputs permutations  $\pi_t(c)$  over candidates  $S_c$ . Exposure  $E_t(s)$  aggregates position-biased visibility with weights  $w_k$ ; outcomes  $Y_t(s)$  are debiased to a merit signal  $q_t(s)$ . A mapping  $g(s)$  induces segments with exposure share  $EP_t(g)$ , compared against targets  $\phi_t(g)$  (merit-proportional or policy frontiers). A fairness-aware allocator feeds constraints back into ranking, while known threats highlight failure modes.

Each item  $s$  can also accumulate success outcomes over time, denoted  $Y_t(s)$  (for example, the number of clicks or transactions item  $s$  has obtained up to time  $t$ ). These outcomes depend on both the item’s intrinsic appeal and the

exposure it has received. Over time, an item that consistently gets more exposure is likely to accumulate more success events, which in turn can feed into the algorithm’s assessment of that item.

We introduce a segment mapping function  $g: S \rightarrow \mathcal{G}$  that assigns each item (or provider) to a segment or group of interest  $\mathcal{G}$  (for example, a group might represent a demographic category, a geographic region, a type of supplier, or any other attribute of concern). Let  $S_g = \{s \in S \mid g(s) = g\}$  be the subset of items belonging to group  $g$ . A central quantity for measuring group fairness in exposure is the per-item exposure share for group  $g$  at time  $t$ , which we define as:

$$EP_t(g) = \frac{1}{|S_g|} \sum_{s \in S_g} E_t(s).$$

This  $EP_t(g)$  represents the average exposure allocated per item in group  $g$  at time  $t$ . It can be interpreted as how much attention an item in group  $g$  typically receives, and it allows us to compare opportunity levels across groups regardless of group size (Jacobides and Lianos 2021). For instance, if  $EP_t(g)$  is significantly lower for a particular group compared to others, it indicates that items in that group are, on average, less visible to consumers.

One goal of a fairness-aware system is to align each group’s exposure share with some notion of the group’s deserved or target exposure. We define a target exposure share for group  $g$ , denoted  $\phi_t(g)$ . This target could be determined in different ways depending on policy. One natural choice is a merit-based target: for example, make  $\phi_t(g)$  proportional to the total estimated quality of items in group  $g$ . Let  $q_t(s)$  be a merit score for item  $s$  at time  $t$  that reflects its underlying quality or relevance (for instance,  $q_t(s)$  could be inferred from debiased historical outcomes via techniques like inverse propensity weighting or other causal adjustments to remove the effect of past exposure). Then a merit-proportional target would set  $\phi_t(g)$  such that  $\phi_t(g) \propto \sum_{s \in S_g} q_t(s)$ , i.e. each group’s share of exposure should be in line with the group’s aggregate merit. Alternatively,  $\phi_t(g)$  might be set by policy or business objectives. This is sometimes referred to as a fairness frontier, representing a deliberate decision about how much opportunity each group should receive relative to others (which could be equal shares, demographic parity, or some bounded affirmative action).

In this framing, algorithmic bias may be characterized as a consistent gap between a group’s actual exposure  $EP_t(g)$  and its aspirational target  $\phi_t(g)$ , which is not explained by actual differences in merit  $q_t(s)$ . Bias can arise and propagate through a variety of pathways. We highlight some of the salient threats and causes of bias in such marketplace platforms: (“Vol. 3, No. 2 (Full Issue)” 2004)

**Historical Selection Bias.** The training data of the ranking algorithm consists of logs of interaction generated by users in the past, which have been influenced by previous exposure decisions. Accordingly, items that were exposed frequently in the past have a large number of recorded interactions, while those rarely exposed have sparse data. Such selection bias implies that the algorithm receives plenty of information about some items—often from the majority groups—and scant information about others—often the minority groups or late entrants. Consequently, the model becomes overconfident for well-exposed items and underconfident for underexposed ones, creating a self-fulfilling prophecy with more exposure given to the former than the latter.

**Context-Group Confounding.** Certain groups might be correlated with particular contexts or features in the data. Suppose, for instance, that group A providers serve urban areas, whereas group B providers serve rural areas. If user behavior varies across such contexts in the data - perhaps because urban consumers have different browsing patterns than rural consumers - then an algorithm might mistakenly capture such contextual differences as proxies for the intrinsic qualities of the groups. Such confounding can cause the system to favor one group in all contexts simply because that group is more prevalent or successful in the dominant contexts in the data.

**Reputation Feedback Loops** (Choi, Nesheim, and Rasul 2015). Most marketplaces use reputation systems that inform ranking through ratings, reviews, and badges. Providers amass high ratings that engender trust and are often ranked higher; their higher ranking creates more customers, hence more opportunities to receive ratings. This creates a snowball effect: Early movers or in-group providers gather reviews quickly and establish a strong reputation, while out-group providers or new entrants struggle to accumulate reviews. The resulting disparity in reputation feeds into the ranking algorithm, which further entrenches the visibility gap.

**Cold Start and Sparse Data.** The “cold start” problem concerns new entrants to the marketplace, or items from groups that have very few representatives. In the absence of historical data on those items, the algorithm may be overcautious in recommending them—so as not to risk a poor user experience. The items therefore remain obscure, generating little data, and the cold start period persists. This is particularly problematic for underrepresented groups, where, even after some time has passed, the sum of all interactions may be sparse, leading to high uncertainty in the model’s estimates for those groups.

**Perception and Trust Gaps.** In addition to measurable gaps, we have the question of user perceptions about the system’s fairness (Welfens, Perret, and Erdem 2010). There may be a gap in perceptions between a platform’s fairness metrics and the sentiments of users or providers. For instance, even when the platform thinks it is being

fair to groups on average, individuals in a minority group might often feel that they are at a disadvantage ("people like me are rarely featured"), so people may feel that the system is biased. The more such perceptions spread through word of mouth or social media, the more they harm the reputation of the platform, independent of measured metrics. The problem of bias in two-sided marketplaces is intertwined with dynamic factors. Our threat model underlines the fact that bias is not a static artifact but an evolving property of the interaction between the system and its users. Any proposed mitigation thus needs to address this dynamism by ensuring that interventions do not only fix a snapshot of the problem but also change the longer-term trajectory of the marketplace toward more equitable outcomes.

### Stage-Wise Framework for Bias Mitigation

The stages are arranged roughly in the chronological order in which they would be implemented or encountered in the development and deployment process. However, these stages are interdependent and form a continuous feedback loop in an operating platform. Addressing bias at multiple junctures—data, modeling, allocation, feedback, and governance—the framework aims to create reinforcing mechanisms of fairness, rather than allowing reinforcing mechanisms of bias. We describe each stage in detail below. (Liu, Wei, and Xiao 2019)

#### Stage 1: Measurement and Ontology Design

Properly measuring bias is the first step toward mitigating it. It involves defining appropriate fairness metrics and developing an ontology of segments (groups) that the platform cares about. In a two-sided marketplace, we elevate exposure and opportunity to first-class metrics alongside traditional performance metrics such as click-through rate or revenue. That is, for any given segment of suppliers or content, the platform explicitly tracks how their exposure compares to desired baselines.

An essential part of this step involves the creation of a strong segment ontology. That is, the platform has to identify what sets of items or suppliers are relevant when monitoring fairness. This could be protected attributes (for instance, gender or ethnicity of a supplier if such information is available and it is lawful to use it), business-relevant categories (region, language, time since joining), or other attributes, such as whether a seller is a small business versus a large merchant. Intersectional groups might also be included in the ontology if those intersections are thought to face special disadvantages. Care should be taken not to carve the data into too many segments, because small sample sizes in each category lead to high-variance measurements. Segments should be chosen based on both the platform’s values and where significant biases could plausibly occur given the marketplace context.

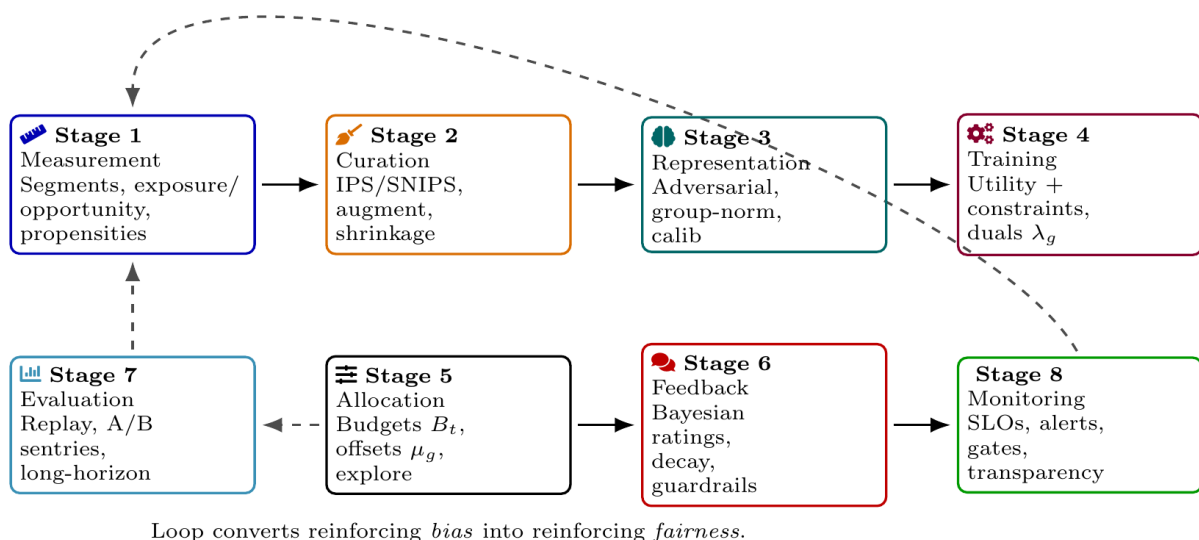
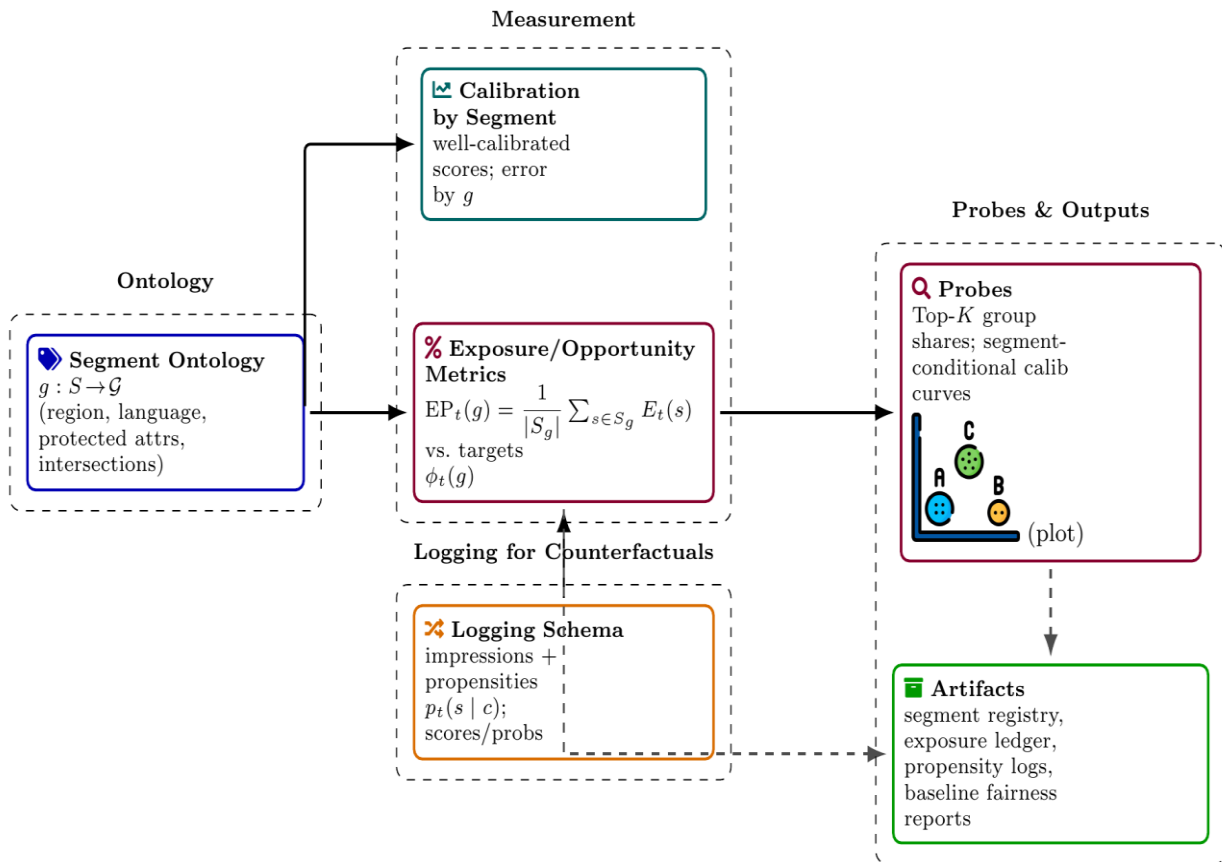


Figure 8: Eight-stage framework.

With segments defined, the platform establishes measurement procedures for fairness metrics. This at least means tracking exposure and success outcomes by segment over time (Süssmuth 2021). As an example, one might continuously compute the average exposure per item  $EP_t(g)$  as defined in Section 2 for each group, and compare it to the group’s target  $\phi_t(g)$ , or to other groups’ exposure. Another important set of metrics involves calibration and accuracy by segment: for instance, checking whether the predicted probabilities of success - say, clicks or

purchases of items - are well-calibrated within each segment, or whether the model systematically overestimates or underestimates the performance of certain groups.



**Figure 9:** Stage 1: Measurement and Ontology Design. Define a segment ontology  $g$ , compute exposure/opportunity metrics and calibration by group, log impressions with propensities  $p_t(s | c)$  for counterfactual evaluation, and maintain probes (e.g., top- $K$  diversity, calibration curves). Outputs include a segment registry, exposure ledger, logging schema, and baseline fairness reports.

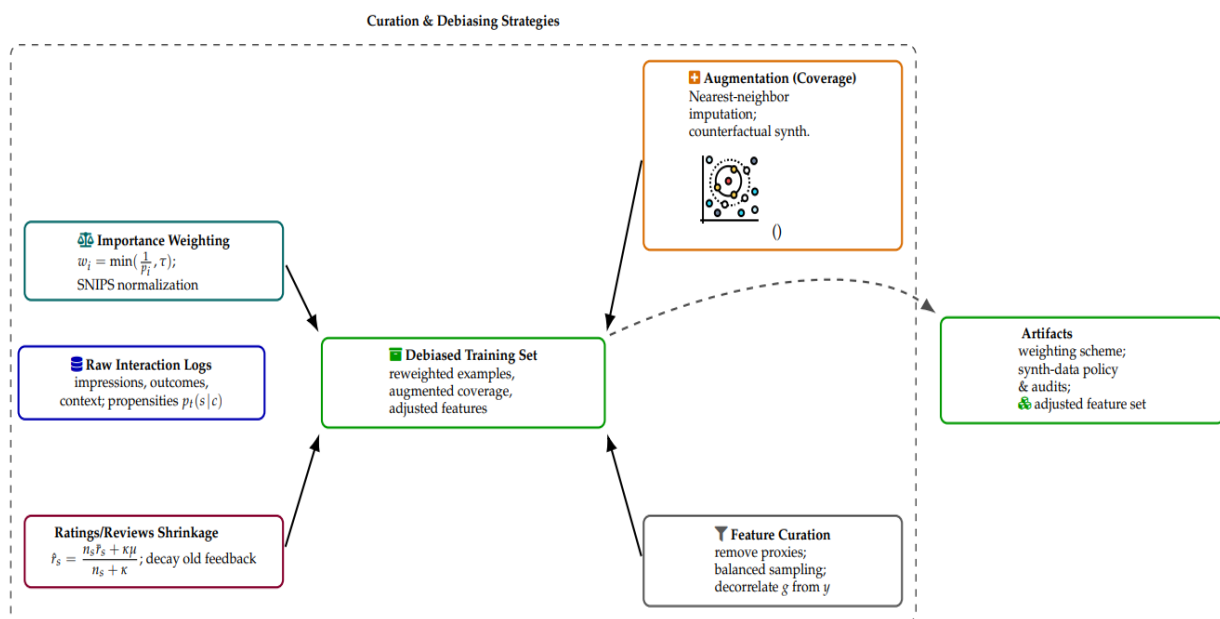
Accurate measurement in a live system also requires proper logging of algorithmic decisions. To analyze and replay the decision-making of ranking algorithms offline—for instance, to try out different algorithms on historical data in a unbiased way—the system should log not only what was shown but also the algorithm’s internal probabilities or scores. In particular, if the production ranker has a stochastic component or an exploration policy, it should log the probability  $p_t(s | c)$  of each item being chosen for request  $c$ . These logged propensities enable counterfactual evaluation: one can later simulate how a different ranking policy would have performed on the same requests by re-weighting outcomes with importance sampling, rather than just observing the biased outcomes from the original policy. This is crucial for unbiased offline measurement of fairness interventions, since it corrects for the fact that some items were rarely shown.

Specialized probes or tests for fairness can be implemented as part of measurement. Segment-conditional calibration curves can be maintained, for instance, as a means to visually inspect if predicted scores versus actual outcomes align on a per-group basis. Another probe might be computing the distribution of impressions among groups for the top  $K$  results of queries (to detect if certain groups almost never appear in top ranks). By codifying these measurement tools and metrics at the outset, the platform creates a baseline understanding of where disparities exist and sets up the quantitative means to detect changes—both positive and negative—in those disparities as new algorithms are deployed. In all, Stage 1 sets up the “sensing” capabilities of the system regarding fairness (Cummings et al. 2019). The major artifacts developed in this stage are the segment registry or taxonomy, an exposure ledger, records of who was shown when and where, a logging schema that captures the necessary data—especially propensities for unbiased analysis—and baseline calibration and fairness reports for each segment. These instruments will guide and validate interventions in later stages.

## Stage 2: Dataset Curation and Debiasing

The next stage involves preparing the training data for machine learning models in such a way that the biases present in the collected data can be lessened. The raw interaction logs from the marketplace reflect historical policy. As a result, they embed the selection biases and feedback loops described in Section 2. Used naively, these data will only teach the model how to echo, and maybe even further sharpen, existing biases. Hence, Stage 2 applies dataset curation and debiasing methods prior to model training.

One basic approach is to employ importance weighting to correct for historical selection bias. Every training example (say, an impression of item  $s$  to a user context  $c$  and whether that item was clicked or not) is reweighted by the inverse of the probability that  $s$  would have been shown in  $c$  under the historical policy. In other words, if the log indicates that item  $s$  was shown to user query  $c$  with probability  $p$ , then that interaction may be given a weight proportional to  $1/p$  in the training loss. This inverse propensity weighting corrects the bias that popular items-with high  $p$  of being shown-are over-represented in the data and that rare items-with low  $p$ -are under-represented. In practice, to avoid extremely large weights for very unlikely events-which can increase variance and destabilize training-one may use a normalized or clipped version of IPS. Techniques like SNIPS self-normalize so that the total weight of examples remains in a reasonable range, reducing variance while still correcting bias (Cambridge University Press, 2017).



**Figure 10.** Stage 2: Dataset Curation and Debiasing. Historical logs with propensities feed importance weighting (with clipping/SNIPS), augmentation for coverage, Bayesian shrinkage for ratings, and feature curation to remove proxies and decorrelate group from outcome. Outputs are a debiased training set plus documented weighting and synthetic-data policies and adjusted features.

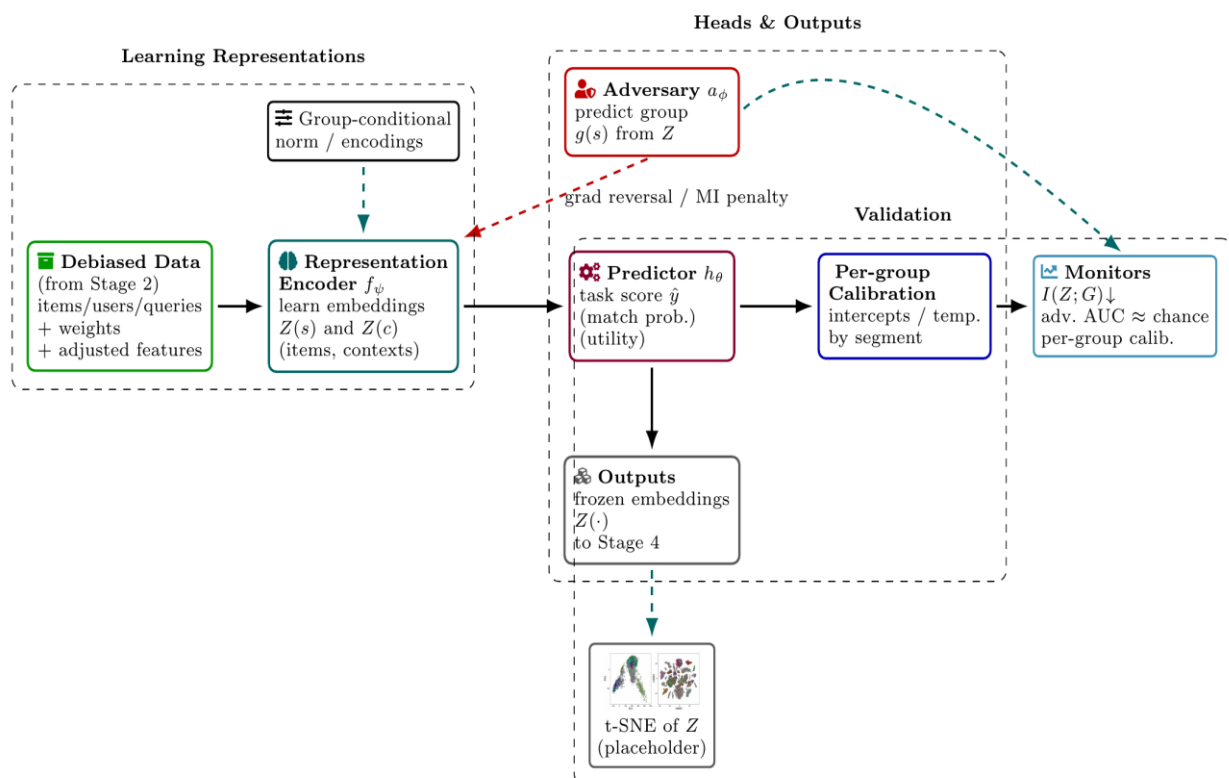
Beyond simple re-weighting of existing data, Stage 2 is dataset augmentation in order to have enough coverage for all segments and contexts. The goal here is to address a common problem where some groups lack (or have sparse) data. This is sometimes colloquially referred to as the “positivity” problem: in many applications, some segment-context pairs have zero or only a few examples in history. If a particular group of providers never showed up in certain types of searches in the past, then the model simply has no signal on how they would perform in that context. As a result, they may be at an unjust disadvantage going forward. To address this problem, the platform can create synthetic counterfactual training examples. One option is nearest neighbor imputation in feature space: find similar contexts or items where the data exists and use their outcomes as a proxy. Another option is the use of generative models to generate plausible interactions (for example: generate the fact that a user saw a minority provider’s listing for a popular query, and did or did not click it, based on what the model thinks would happen). It is crucial that any such synthetically generated examples are created carefully in order to remain realistic and not introduce artifacts; they must in general be bounded by plausibility constraints.

In collecting the dataset, we also pay attention to how existing performance signals like ratings and reviews are incorporated. If raw ratings are taken at face value, items from groups with fewer ratings-or systematically biased ratings-could be misjudged. One debiasing approach here is to use hierarchical shrinkage or Bayesian estimation for such signals. For instance, instead of using an item’s average star rating directly-e.g., 5.0 if it has only one review while an established item receives 4.5 from 100 reviews and is thus most likely reliably good-we move each item’s estimated quality toward the global average based on a signal’s statistical confidence. This way, items

from minority groups that may have a low count of reviews or erratic early feedback will not be unfairly penalized or boosted by small sample noise. The model should basically understand that a 5-star rating with one review is in no way better than a 4.5 with many reviews; it may treat the former more as an unknown with a slight indication of being positive (Hagiu and Yoffie 2013). Curation also involves the removal or mitigation of any spurious correlations that may cause the model to learn discriminatory patterns. Suppose, for example, that group membership can be inferred from certain features—such as the location, or an image—and that correlates with outcome due to historical bias; one might either remove those features or decorrelate them in the training data, by balanced sampling for instance. The goal is not to blind the model to features that are legitimately relevant, but to prevent it from latching onto group identity as a proxy for something else in a way that hurts fairness. By the end of Stage 2, the training dataset has been reweighted, augmented, and filtered to the extent possible to reflect a more fair distribution of examples. It now contains information that gives each segment a fighting chance for the model to learn its true quality signals, rather than just reinforcing past exposure patterns. Key artifacts from this stage include the weighting scheme, or weights attached to each training example; documentation of any synthetic data generation policies, with transparency and audit logs to ensure these augmentations are interpretable and justifiable; and the adjusted input features, such as shrinkage-adjusted ratings. These will feed into the model training stage that follows.

### Stage 3: Fair Representation Learning

It is often very useful, before training a marketplace ranking model on the prepared data, to learn intermediate representations of items and users that are helpful for prediction. We focus on learning these representations in Stage 3 so as to encourage fairness and minimize the encoding of spurious biases. The goal is to obtain feature embeddings or latent representations for items—and possibly users or queries—that capture the relevant information useful for matching supply and demand, while filtering out or neutralizing information that would lead to unjustified bias against certain groups.



**Figure 11.** Stage 3: Fair Representation Learning. A representation encoder  $f_\psi$  (with group-conditional normalization) produces embeddings  $Z$  that feed a task predictor  $h_\theta$  and an adversary  $a_\phi$  that tries to infer  $g(s)$ . A gradient-reversal/MI penalty discourages group leakage in  $Z$ . Per-group calibration layers adjust scores. Outputs are frozen embeddings for Stage 4, while monitors track  $I(Z; G)$ , adversary AUC, and calibration by segment.

One approach is called adversarial representation learning (Ribeiro-Navarrete et al. 2021). Here, in addition to our main prediction task (say, predicting the likelihood of a match between an item and a consumer), we jointly train an adversary network whose objective is to predict the protected group or segment  $g(s)$  of the item from the learned representation. Representation is learned by a neural network (or any parametric) that feeds into both the main predictor and the adversary. During training, the representation learner is optimized not only to help the main

task-e.g., maximize prediction accuracy or utility-but also to confuse the adversary. In practice, this can be implemented by reversing the gradient from the adversary or adding a term to the loss that penalizes mutual information between the representation and the group attribute. The intended outcome is that, to the extent possible, the representation  $Z(s)$  of an item  $s$  contains no easy-to-extract information about the item's group membership beyond what is inherently needed for the main task. For example, if two providers are identical in every way except for their group, their representations should be very similar, allowing the model to treat them similarly at inference time.

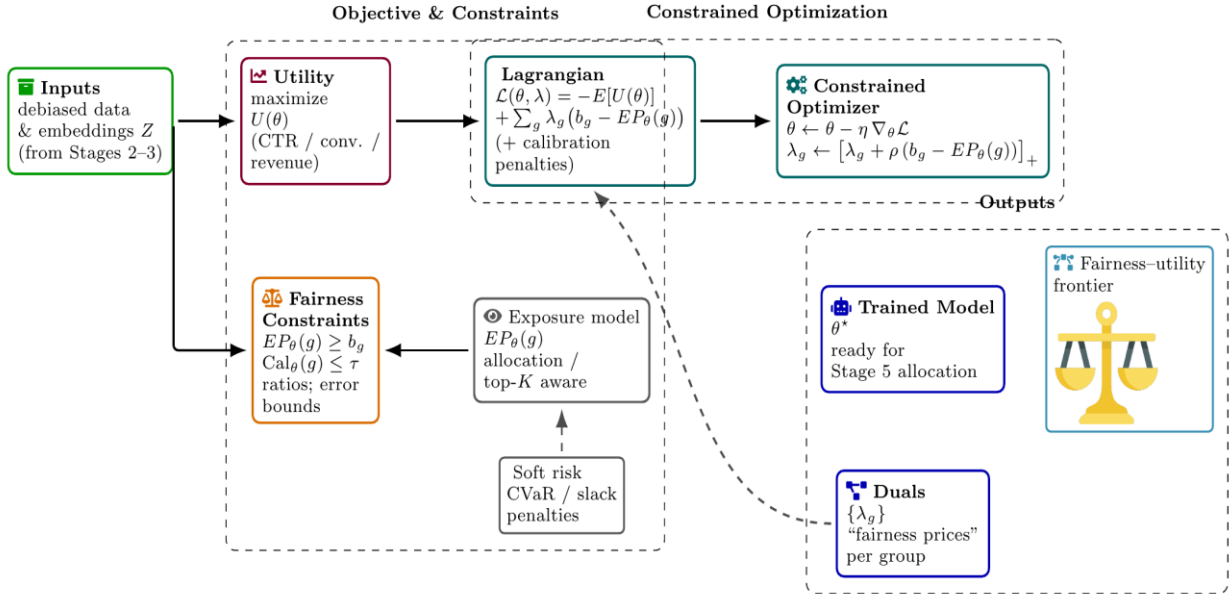
Another approach is the use of group-conditional normalization or encoding strategies. In neural network models using batch normalization layers, for example, the statistics (mean and variance) can be computed separately for each group rather than across the whole batch. This can be important because if one group dominates a batch, the batch statistics may not be representative for the distribution of a minority group, which may lead to inappropriate normalization of that group's data and hurt its performance. The intuition behind doing batch normalization within each segment-in cases when enough data per segment exists-is to ensure that each group's features are scaled in a manner appropriate to that group's range of feature values. This can help avoid a subtle form of leakage where global normalization constants inadvertently carry information about group identity-for instance, because the overall mean of a feature is different largely due to one group's prevalence.

Finally, the model can incorporate per-segment calibration. If we have information that, in the wake of initial training, the scores produced by the model – e.g., predicted probability of successful match – are systematically too high for one group and too low for another, then we can introduce calibration layers or parameters that shift or scale the outputs for each segment (Naudé 2022). Those might be simple intercept adjustments or temperature scaling factors learned on a validation set to ensure that, say, a 0.8 predicted probability represents the same likelihood of success for items in group  $A$  as it does for those in group  $B$ . In this way we may ensure that no group is consistently over-predicted or under-predicted, which is important for fairness when those scores are used to determine rankings.

Throughout representation learning, we prefer fairness criteria that emphasize sufficiency rather than forcing demographic parity in the representations. Sufficiency, in this case, is defined as the characteristic that, conditioned on the model's score or representation, the outcome is independent of the group-in other words, the model's predictions are equally valid for all groups. Sufficiency avoids carrying an unnecessary performance gap between groups. On the other hand, demographic parity at the representation level would imply that each group's representations follow the same distribution, which might be too strong and may impede the model's ability to distinguish truly relevant differences-for example, if group  $A$  legitimately specializes in a category that has higher conversion rates, the representation should capture that without being forced to look like group  $B$ 's distribution. To validate our representation learning stage, we track metrics including the mutual information  $I(Z; G)$  between the learned item representation  $Z$  and the group variable  $G$ . Low mutual information-close to zero-means that from the representation alone, there is little information about which group an item originated from, indicating that the de-correlation of the representation from the protected attribute has been successful. We also look at the performance of the adversary model: it should be unable to do much better than chance in its prediction of group membership given the representation. At the same time, we verify that the performance of the main task remains high and that calibration within each group is improving. The output from Stage 3 consists of a set of learned item- and probably user/query-representations that feed into the final ranking or matching model. These representations have already been shaped to minimize encoding of bias, forming a more equitable basis for this next step, where the model will be trained with explicit fairness objectives. (Ellerman 2014)

#### **Stage 4: Objective Design and Constrained Model Training**

With debiased data and fair representations in hand, Stage 4 tackles the core machine learning task: training the ranking/matching model under a multi-objective criterion that includes both utility (performance) and fairness constraints. Rather than optimizing a single metric like predictive accuracy or revenue, we explicitly encode fairness requirements into the objective function or as constraints alongside the objective.



**Figure 12.** Stage 4: Objective design and constrained training. Utility  $U(\theta)$  is optimized under fairness constraints (exposure  $EP_\theta(g)$  and calibration). A Lagrangian with duals  $\lambda_g$  drives joint updates of model parameters and constraint multipliers. Outputs are the trained model  $\theta^*$  learned duals (interpretable as fairness prices), and a fairness-utility frontier for policy selection.

Formally, let  $U(\theta)$  denote the expected utility that the platform aims to maximize, as a function of the model parameters  $\theta$ . This utility could be a combination of metrics such as overall click-through rate, conversion rate, or revenue generated by the recommendations (possibly minus any penalties for latency or other costs). We then introduce constraints to enforce fairness goals, for example, for each group  $g \in \mathcal{G}$ ,  $EP_\theta(g) \geq b_g$  and  $Cal_\theta(g) \leq \tau$ , where  $EP_\theta(g)$  is the modeled or expected exposure share for group  $g$  under the model  $\theta$  (we include the subscript  $\theta$  to indicate it depends on the model’s predictions and subsequent allocation mechanism),  $b_g$  is a minimum acceptable exposure level for group  $g$  (potentially based on the target  $\phi(g)$  or a fraction thereof), and  $Cal_\theta(g)$  is a measure of miscalibration for group  $g$  (for instance, the difference between predicted and actual success probabilities for items in  $g$ , or some error metric that should be below a threshold  $\tau$ ).

These constraints formalize fairness criteria like “each group should get at least a certain level of exposure relative to its size or merit” and “the model should be well-calibrated for each group.” Other types of constraints could also be incorporated, such as bounds on between-group differences (e.g.,  $EP_\theta(A)/EP_\theta(B)$  should not exceed some ratio), or constraints on error rates if it were a classification task (like equal false positive rates across groups). The framework is flexible to the specific choice of fairness definitions as needed by the platform’s policy.

To solve this constrained optimization problem, we can use the method of Lagrange multipliers and iterative updates. We construct a Lagrangian:

$$\mathcal{L}(\theta, \{\lambda_g\}) = -\mathbb{E}[U(\theta)] + \sum_{g \in \mathcal{G}} \lambda_g (b_g - EP_\theta(g)).$$

Here  $\lambda_g \geq 0$  are Lagrange multipliers (dual variables) for the exposure constraints. We incorporate calibration constraints similarly, either with their own multipliers or by converting them into a differentiable penalty in the objective if direct enforcement is tricky. Intuitively, if a particular group  $g$  is not getting enough exposure under the current model (i.e. (Goldfarb and Tucker 2019).  $EP_\theta(g) < b_g$ ), then in the gradient step the term  $\lambda_g (b_g - EP_\theta(g))$  will increase (as  $\lambda_g$  is typically adjusted upwards when constraints are violated), effectively pushing the model to increase that group’s scores or otherwise alter allocations to raise their exposure. Conversely, if a group is exceeding its minimum, the multiplier may stay low or zero and not interfere much with the utility optimization.

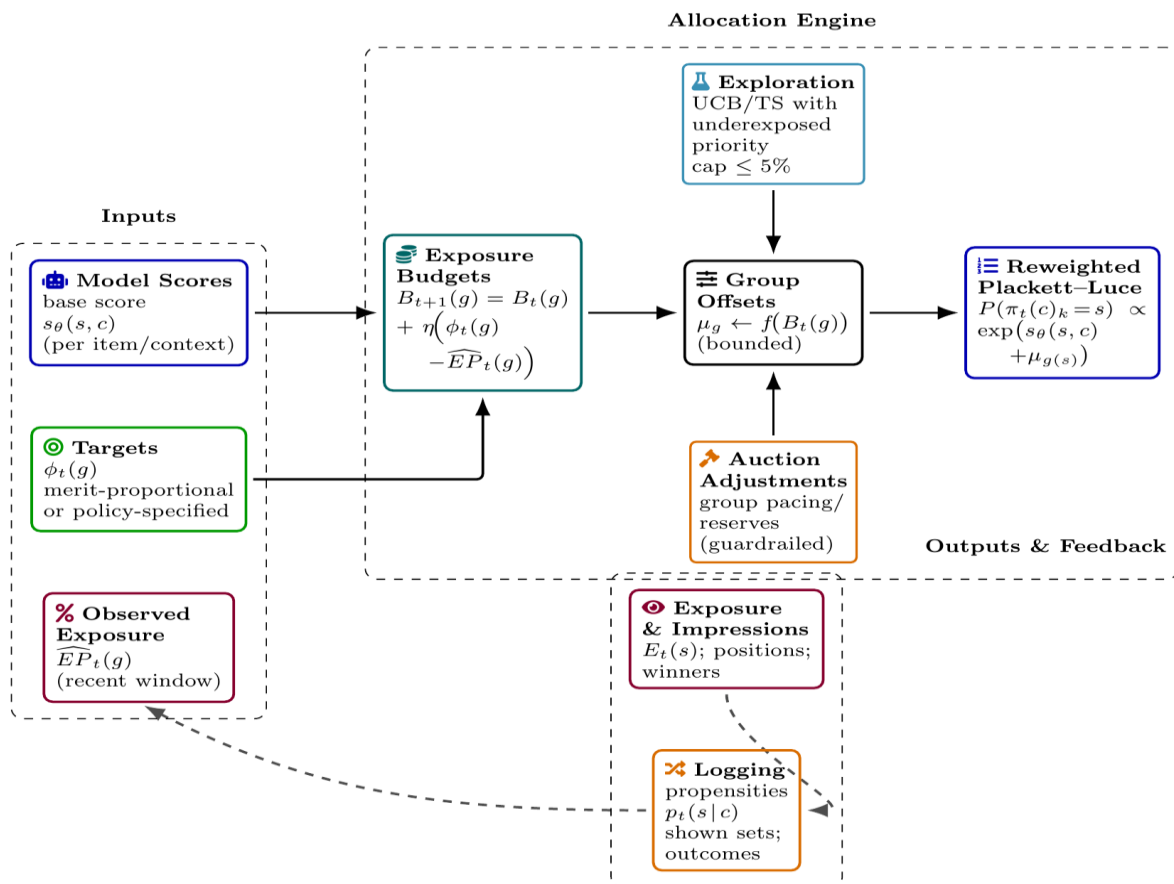
One has to be careful in updating the dual variables in a stable manner; this can be done through subgradient methods or adaptive heuristics during training. If the constraints are tight-meaning, the optimal solution lies precisely on the boundary of the fairness limits-the training can oscillate: overshooting the constraint then correcting back. In order to mitigate such oscillations, it is possible to add a small quadratic penalty or to use techniques from constrained optimization that penalize large changes of (which is akin to adding curvature to the

dual problem). Another practical approach is to gradually ramp up the enforcement of fairness constraints—start by setting and gradually increase the penalties during training—so the model has time to adjust without diverging.

In some cases, the platform might allow soft constraints or trade-offs rather than absolute hard constraints. For example, instead of requiring  $EP_\theta(g) \geq b_g$  with no exceptions, the platform might say “minimize the risk that any group’s exposure falls too far below target.” In such a case, one could incorporate a penalty that heavily punishes outcomes where a group’s exposure is in the worst  $q$ -quantile of the distribution (this is analogous to a Conditional Value at Risk (CVaR) formulation, treating low exposure for a group as a “loss” whose expectation in the tail we want to bound). Another approach is to allow slack variables in constraints with an overall penalty for using that slack (so occasional mild violations are permitted but carry a cost in the objective). These formulations can provide a safety valve so that if strict fairness would drastically hurt utility, the system can accept a slight constraint violation and pay a known cost for it. The result of Stage 4 is a trained model (or several models if ensemble) that intrinsically balances utility and fairness (Ugolini 2021). Importantly, the model is accompanied by the final values of the dual variables—the  $\lambda_g$  for each constraint. These can be interpreted: for example, a high  $\lambda_g$  at optimum indicates that the model had to sacrifice a lot of utility to satisfy group  $g$ ’s exposure requirement, with the implication that  $g$  was a particularly disadvantaged group under unconstrained optimization. These duals become important for the next stage, as they will inform how the online system should prioritize certain groups; they essentially indicate the “price” of fairness for each group. We thus obtain, along with the model, a set of fairness constraint parameters that will be used to guide real-time allocation.

### Stage 5: Fair Allocation and Ranking Mechanism

With a model capable of scoring or ranking items, Stage 5 addresses how to distribute exposure in the live marketplace such that the fairness constraints and objectives are upheld in real-time. The output of the model, e.g., a relevance score or predicted probability of success for each item in a given context, is not necessarily the final say in what gets shown. We introduce a fairness-aware allocation layer on top of the model’s scores. This layer uses the notion of exposure budgets for each segment in order to selectively modify ranking or selection probabilities in favor of under-served groups when necessary, while still paying heed to user experience through gradual changes and exploration.



**Figure 13.** Stage 5: Fair Allocation and Ranking. Exposure budgets  $B_t(g)$  update from targets  $\phi_t(g)$  and observed exposure  $EP_t(g)$ . Budgets induce group offsets  $\mu_g$  used in a reweighted Plackett–Luce ranking; exploration and (optional) auction adjustments act as sidecars. Resulting exposure  $E_t(s)$  and propensities  $p_t(s | c)$  are logged and fed back to maintain targets without degrading relevance.

For each group  $g$ , we maintain an exposure budget  $B_t(g)$  at time  $t$ . This budget is a measure of how much more (or less) exposure the group should receive relative to the current trend to meet its target  $\phi_t(g)$ . The intuition is as follows: if a group is getting less than its target, then it would have a positive budget, indicating that it is "owed" exposure; if it is getting more, the budget might be zero or negative, indicating it should be throttled so as not to greatly exceed targets. The budgets are updated over time with a small learning rate or pacer parameter  $\eta$  (Ribeiro and Bao 2021). One simple update rule at discrete intervals (say each day or each batch of interactions) could be given by

$$B_{t+1}(g) = B_t(g) + \eta \left( \phi_t(g) - \widehat{EP}_t(g) \right).$$

Here  $\widehat{EP}_t(g)$  is the observed exposure share for group  $g$  in the recent window (for instance, the last hour or day). The term  $\phi_t(g) - \widehat{EP}_t(g)$  is the gap between desired and actual exposure. If  $\widehat{EP}_t(g)$  fell short of  $\phi_t(g)$ ,  $B_{t+1}(g)$  increases a bit, meaning we plan to give more boost to group  $g$ ; if  $\widehat{EP}_t(g)$  overshoot,  $B_{t+1}(g)$  might decrease. The parameter  $\eta$  is chosen to be small (e.g. a fraction like 0.01), so that adjustments are gradual rather than abrupt. This avoids jarring changes in the marketplace that could confuse users or providers.

How do budgets affect the actual ranking of items for a request? We incorporate them via score offsets or adjustments. Specifically, when the model produces a base score  $s_\theta(s, c)$  for an item  $s$  in context  $c$ , we adjust this score by an amount related to the item's group. One flexible approach is to use a probabilistic ranking mechanism such as a reweighted Plackett–Luce model. Under Plackett–Luce, the probability of a particular item being chosen at rank  $k$  (given that some items have already been chosen for higher ranks) is proportional to an exponential of its score. We modify this as:

$$P(\pi_t(c)_k = s \mid \pi_t(c)_{1:k-1}) \propto \exp(s_\theta(s, c) + \mu_{g(s)}),$$

where  $\mu_{g(s)}$  is an offset (which can be positive, negative, or zero) corresponding to the group of item  $s$ . Essentially, each group  $g$  has a bias term  $\mu_g$  that gets added to all items of that group when deciding the probabilities for ranking. These  $\mu_g$  terms are tuned such that if a group is under its exposure target,  $\mu_g$  will be higher (boosting items from that group in the rankings probabilistically), and if a group is over its target,  $\mu_g$  might be zero or negative (slightly demoting items to prevent excessive exposure beyond target). The exact values of  $\mu_g$  can be learned or updated based on the budgets  $B_t(g)$  for example, one could set  $\mu_g = f(B_t(g))$  for some increasing function  $f$ , or even run a small optimization at each update to find  $\mu_g$  that would equalize EP to  $\phi$  given current traffic patterns. (Blanchet and Fleurbaey 2020)

Importantly, this approach still respects user relevance to a large degree. The base score  $s_\theta(s, c)$  ensures that high-quality matches are ranked highly. The group offset  $\mu_{g(s)}$  only nudges the ranking probabilistically. If an item from a disadvantaged group is nearly as relevant as a top item from an advantaged group, the offset might cause it to occasionally leapfrog and get the top slot, whereas without the offset it might always remain second. Over many requests, these small nudges accumulate to significant increases in that group's total exposure, helping to fulfill the fairness goals, but any given user request is not dramatically altered to the point of showing irrelevant content.

Another dimension of Stage 5 is incorporating exploration deliberately for fairness. Exploration in recommender systems or marketplaces means occasionally showing items of uncertain quality to gather more data about them (trading off immediate optimality for long-term learning). Here, we align exploration with fairness by ensuring that underexposed groups are given some priority in exploration. For example, using a Thompson Sampling or Upper Confidence Bound (UCB) approach, we can maintain uncertainty estimates for each item or group and allocate a small percentage of impressions to items that have high uncertainty (which often will include items from minority or new groups due to lack of data). By doing so, we not only treat those items more fairly in the short run, but we also reduce uncertainty about them faster, which can correct the model's underestimation if in fact those items are high quality. The exploration policy should be variance-controlled; in other words, we might constrain it such that the fraction of exploratory (less certain) content never exceeds, say, 5% of total impressions, to manage risk. The idea is to systematically give chances to content that might otherwise be perpetually sidelined due to lack of data, thereby breaking the vicious cycle.

If the marketplace uses an auction-based mechanism (as in ad marketplaces or certain gig marketplaces where providers "bid" or set prices), fairness interventions can be implemented via adjustments like group-specific reserve prices or pacing of bids. For instance, the system might impose a slightly lower reserve price (minimum bid to win an impression) for items from an underrepresented group, effectively giving them a better chance to win in the auction despite perhaps having slightly lower predicted conversion. Alternatively, the platform itself can act as a bidder that boosts certain groups. It can spend some exposure "budget" to ensure they appear. In doing so, strict guardrails are needed: the system should avoid cross-subsidizing to an extent that severely hurts overall efficiency or user trust. If, for example, items from a low-quality group were boosted too much, users might start ignoring recommendations or leaving the platform, which is a negative outcome for all. Hence, any such multiplier or reserve adjustment would be bounded and closely monitored.

In summary, Stage 5 translates the fairness-aware model into real-time decision rules for what to show. It is the enforcement arm of fairness during live operations. Key components from this stage include the budgeting mechanism that tracks exposure debts/credits for each group, the adjusted ranking algorithm that integrates these budgets (through score offsets like  $\mu_g$  or equivalent techniques), and the exploration policy tuned for reducing bias. All of these must be implemented efficiently and reliably, typically within the serving infrastructure of the marketplace, and are accompanied by safety checks to ensure they do not degrade the core matching quality beyond acceptable limits.

### Stage 6: Feedback Shaping and Reputation Dynamics

After the allocation and interactions have taken place, the marketplace collects feedback in various forms: ratings, reviews, likes, complaints, etc. Stage 6 focuses on how to shape and use this feedback in a way that avoids reinforcing bias and instead helps attenuate it (Gamper 2012). The reputation system-together with how user feedback is solicited, computed, and fed back into rankings-may exacerbate or help correct existing disparities.

Table 1. Feedback Design and Adjustment Mechanisms

Aspect	Issue / Bias Source	Adjustment Method	Intended Effect
Runaway reputations	Early feedback dominates rankings; low-review providers disadvantaged	Bayesian estimators with global priors and time decay	Stabilize scores; allow recovery and fair comparison
Visibility distortion	High clicks from position bias inflate perceived quality	Separate service-quality vs. visibility-driven feedback	Correct for exposure effects in reputation computation
Temporal bias	Old reviews overweighted in cumulative rating	Decay old feedback contributions	Reflect provider improvement or change over time
Feedback heterogeneity	Ratings capture context misfit, not true quality	Context-aware modeling of feedback signals	Attribute blame correctly between item and placement

Table 2. Reputation Integration, Fairness, and Model Guardrails

Mechanism	Implementation Detail	Effect on Platform Dynamics	Fairness Objective
Uniform solicitation	Standardized review request frequency across segments	Prevents feedback-rate bias between provider groups	Equal opportunity for feedback visibility
Gradient throttling	Cap model sensitivity to reputation-based features	Avoids overreaction to small rating differences	Reduce rich-get-richer amplification
Flattened reputation curve	Diminishing marginal impact beyond threshold	Enables newcomers to catch up post-credibility phase	Promote competitive equity
Feedback loop correction	Feed adjusted reputation into training data	Stabilizes learning and ranking fairness	Maintain unbiased merit estimates $q_t(s)$ over time

One aspect of designing feedback is the use of statistical techniques to make reputation measures fairer and more informative. Straightforward uses of average ratings or cumulative review counts can be misleading. For instance, a provider who has 50 five-star reviews may outrank a new provider who has only 2 reviews, even if that new provider could potentially give as good a service. In order to avoid runaway reputations, we employ Bayesian estimators with priors and decays. For each item or provider, rather than take the raw average rating, we compute a posterior estimate of quality that starts with a prior-e.g., a global average rating as the prior-updated with each new review. This implies that a provider with a few reviews will have an estimated rating pulled toward the global average rather than swinging to extremes. We might also decay the influence of very old feedback over time, under an assumption that providers are able to improve or change. A bad review several years old might count less than more recent reviews. These measures prevent early success or early missteps from permanently dictating an item’s standing.

Another aspect is the separation of various kinds of feedback signals. User actions and feedback can be influenced by many factors not purely related to item quality. For example, an item might receive a low rating not because it was objectively poor, but because the user’s expectations were misaligned or because it was shown in a context in which it was a poor fit-which is arguably the algorithm’s mistake rather than the item’s (“PAPERS FROM ACTUARIAL JOURNALS WORLDWIDE” 2016). We therefore distinguish between service quality signals and

visibility-driven signals. Service quality might be gleaned from detailed reviews or return behavior-did the user rebook the same provider, etc.-whereas engagement-clicks, likes-might be largely driven by visibility: top-ranked items naturally get more clicks regardless of quality. By modeling these separately, the platform can avoid giving too much credit to items just because they were placed in favorable positions. For example, we might down-weight the contribution of clicks to reputation for items which were always shown in the first position since their high click rate is partly due to position bias.

It is also important to consider the consistency in how feedback is being elicited. The platform needs to have the same or, at least, equivalent solicitation policies across all segments. If the platform sends reminders via email to request a review from consumers, it needs to be sending these reminders consistently; it cannot ask more often - even unwittingly - when the provider belongs to a particular group, just because those providers happen to be in scenarios where a follow-up gets triggered more frequently. Any discrepancy in the probability of feedback can create distortion in apparent group performance. For example, if customers tend to leave reviews more frequently for providers of group A than for group B, due to a psychological bias or for certain differences in engagement, the system should be aware of that and compensate, or - even better - design the interface to have equal rates of feedback. When reinjecting reputation signals into the ranking model or the allocation logic, we introduce guardrails such as gradient throttling for cumulative reputation. In model training, when a feature input corresponds to an item's reputation-such as average rating or number of past sales-we can cap the magnitude of the model weights on that feature or otherwise regularize it. This means the model will not overreact to a small difference in reputation, especially for items with vastly different amounts of information. Another strategy is to flatten the curve of returns on reputation, such that the difference between 0 and 10 reviews might be treated as more important than the difference between 100 and 110 reviews (diminishing returns), to give newcomers a chance to catch up once they clear an initial credibility threshold. These feedback-shaping practices have the net effect of preventing a rich-get-richer scenario driven by early exposure advantages alone (Havrylchuk and Verdier 2018). We want the feedback to reflect true quality to the extent possible, rather than just historical visibility. By controlling how feedback is aggregated and used, the platform can slow the rate at which small early differences turn into huge gaps. This in turn buys time and opportunity for high-quality providers from any group to surface eventually, even if they start off with less visibility. Stage 6 thus feeds back into Stage 2 and Stage 4 in a virtuous cycle: the adjusted feedback becomes part of the data used for model training, and it ensures that our earlier assumptions-like the  $q_t(s)$  merit estimates-remain as unbiased as possible over time.

### Stage 7: Evaluation and Continuous Improvement

After these mechanisms are implemented within a model and system, proper effect evaluation should be carefully carried out in advance of, and during, its deployment. Stage 7 covers how offline simulations, online experiments, and long-term monitoring should be conducted to ensure that the fairness mechanisms achieve their desired effects and further calibrate the trade-offs between fairness and other objectives.

Table 3. Evaluation Framework for Fairness Mechanisms (Stage 7)

Evaluation Mode	Description / Objective	Key Metrics	Intended Insight
Offline replay evaluation	Simulate new model using historical logs and propensities	Total clicks/revenue, EP(g), opportunity lift, calibration error	Estimate policy effects pre-deployment
Fairness-utility frontier	Vary fairness constraint strength; compare outcomes	Pareto curve between fairness gain and efficiency loss	Quantify trade-offs and select optimal balance
Online A/B testing	Compare live treatment vs. control algorithms	Exposure parity, conversion rates, calibration stability	Validate fairness improvements in production
Interleaving / exposure budgets	Ensure no group loses exposure in test buckets	Group exposure ratios, exposure-at-K	Protect users/providers during experiment
Sentry checks	Automated triggers for fairness degradation	Thresholds on calibration gap or exposure drop	Real-time safeguard against regressions

Offline replay evaluation. Before deploying changes to the live platform, we use historical data to simulate how the new system would have performed. Using the logged propensities and context from Stage 1, we can perform counterfactual analysis: for each past request, re-score the candidates with the new model and fairness adjustments, and then use importance sampling to estimate what the outcomes (clicks, conversions) and exposure allocations would have been under that policy. This gives us offline estimates of key metrics: overall utility (e.g., total clicks or revenue), and fairness metrics such as the exposure EP(g) for each group, the opportunity improvement for the worst-off group, and the calibration error per group. We pay special attention to metrics like textitopportunity lift (the increase in exposure or success probability for historically disadvantaged groups under the new model vs. the old model), and segment-

conditional AUC or accuracy (to ensure the model’s prediction quality is high for all segments). (Ljungqvist, Marston, and Wilhelm 2006)

Table 4. Long-Term Simulation and System Health Monitoring

Simulation Element	Modeling Approach	Observed Indicators	Interpretation / Use
Dynamic platform behavior	Agent-based or MDP simulation of users and providers	Retention, entry/exit dynamics, fairness exposure	Assess sustainability of fairness interventions
Marketplace health index	$H_t = \alpha \cdot \text{Revenue}_t + \beta \cdot \text{Diversity}_t + \gamma \cdot \text{Trust}_t$	Composite of revenue, diversity, and trust	Track multi-objective system evolution
Fairness–revenue trajectory	Compare fairness and utility components over time	Long-horizon changes in $H_t$ components	Detect long-run trade-offs or synergies
Policy iteration	Adjust fairness parameters iteratively in simulation	Convergence to stable equilibrium behavior	Optimize fairness targets dynamically
Reporting artifacts	Offline/online reports and simulation outcomes	Evaluation logs, frontiers, dashboards	Guide deployment decisions and parameter tuning

Often we are deriving fairness-utility frontiers from offline analysis by varying the strength of constraints or fairness parameters. Consider, for example, training multiple models with different minimum exposure requirements  $bg$ , or with a different weight on fairness in the objective, and then evaluating each model offline. This results in a Pareto curve where one axis is a fairness measure such as the lowest group’s exposure or success rate, and the other axis is an overall performance measure such as total clicks or revenue. The resulting frontier can help stakeholders understand the trade-off: how much overall efficiency might we be trading for more fairness, and is that trade-off worthwhile or acceptable? The goal is to choose a point on this frontier that represents a good balance, or to demonstrate that improving fairness does not drastically hurt efficiency in our case (which would strengthen the case for deployment). Interleaving and shadow testing. When performing online A/B testing, special procedures ensure the fairness constraints are still satisfied in each experimental bucket. That is, if we do a standard A/B split-some users see the new algorithm, some see the old-there is a risk that one bucket inadvertently obtains a worse fairness outcome during the test, which would harm the affected users or providers. One strategy for mitigation is interleaving: within the experiment, the ranked results of the new algorithm can be interwoven with the ranked results of the old algorithm in each request in a controlled way so that no group is entirely deprived of exposure in either bucket. Another strategy is to impose exposure budgets within each experimental bucket separately, with the goal that both the control and the treatment maintain fairness thresholds-higher for the treatment perhaps, but the control maintains baseline fairness that existed. These techniques ensure the experiment itself does not create a fairness regression for the sake of measurement. As part of live experiment monitoring, we also set up sentry checks.

These are automated triggers that will stop or revert the experiment if certain fairness metrics degrade beyond a set threshold (Chen and Rizzo 2010). As an example, if we detect that in the treatment bucket, calibration of predicted scores for group X has drifted significantly-say, the predicted probability versus actual outcome gap has grown by more than  $\delta$  for that group-we treat it as a signal that something is not right and stop the experiment. If any group experiences a sharp drop in exposure-that is, group Y’s average exposure per item falls more than  $x\%$  compared to its historical baseline, or compared to the control bucket for over a day-the experiment is stopped. These guardrails protect against unintended consequences that may not immediately appear in overall metrics but can be very detrimental to a subset of users. Long-horizon simulation: The cumulative, long-term impact of deploying a new algorithm might not be captured by short-term tests and offline evaluations. For this reason, we perform simulations or use analytical models in order to project the impact over a longer horizon. We model the marketplace as a dynamic system, potentially as a Markov Decision Process or an agent-based simulation, where providers and consumers enter or quit, and their behavior-having to do with continuing to use the platform or not-is influenced in fairness-related ways by outcomes on the platform. Perhaps a provider has a certain probability of quitting the platform if they don’t get any booking for a long time, where the probability depends on whether the algorithm gave them fair exposure. We simulate user and provider populations whose behaviors are specified in such a way, and we see through these simulations if a fairness intervention yields better retention of diverse providers and therefore a more robust supply in the long run, or if, instead, it creates any market distortions.

We track composite metrics over these simulations, such as a marketplace health index:

$$H_t = \alpha \cdot \text{Revenue}_t + \beta \cdot \text{Diversity}_t + \gamma \cdot \text{Trust}_t,$$

where  $\text{Revenue}_t$  could be the platform’s revenue at time  $t$ ,  $\text{Diversity}_t$  could be a measure of how balanced the outcomes are across groups (for example, entropy of the distribution of conversions across groups, or the minimum group outcome as a fraction of the maximum), and  $\text{Trust}_t$  could be a measure derived from simulated user/provider satisfaction (perhaps proxied by their retention rates or a survey score if we have real data). The weights  $\alpha, \beta, \gamma$  reflect the platform’s priorities. Through these simulations, we can compare the long-term health index for scenarios with and without the fairness framework. Ideally, a successful fairness intervention would show higher trust and diversity components of  $H_t$  over time without a catastrophic drop in revenue component (Chaudhuri 2016).

Key artifacts from Stage 7 include the offline evaluation reports (with metrics and frontiers), the configured automated monitors for experiments, and the simulation results analyzing long-term dynamics. This stage not only validates the system before full deployment but also provides insights to tweak parameters. For instance, if offline tests show that one group still lags, we might tighten its constraint; if simulations reveal that overly aggressive fairness can reduce overall participation after a year, we might moderate the targets or the pace.

### Stage 8: Monitoring, Governance, and Transparency

The final stage recognizes that fairness is not a one-time project but an ongoing commitment. After the system is deployed, continuous monitoring and proper governance structures are needed to maintain fairness over time and to adapt to new patterns in the marketplace. Moreover, transparency with users and other stakeholders becomes crucial for managing perceptions of fairness.

Table 5. Monitoring and Governance Framework for Sustained Fairness (Stage 8)

Component	Function / Purpose	Mechanism / Example	Outcome
Fairness dashboards	Continuous visibility into fairness metrics	Track $EP_t(g)$ , $\phi(g)$ , calibration error with alerts	Early detection of bias or drift
Change detection	Identify sustained deviations from fairness targets	CUSUM, drift tests, confidence-band monitoring	Automated alerts for fairness degradation
Governance gates	Fairness checks in approval workflows	Require fairness parity before model launch	Institutionalized fairness accountability
Incident runbook	Structured response to fairness incidents	Pause updates, activate fallback model, notify teams	Rapid mitigation of emerging bias
Dual-variable freeze	Stabilize fairness parameters under abnormal behavior	Temporarily fix fairness budgets $\lambda_g$	Prevent oscillation or divergence in allocation logic

Table 6. Transparency, Communication, and Accountability Measures

Dimension	Implementation	Stakeholder Impact	Goal
User-facing transparency	Explainability tools and fair opportunity messaging	Providers understand exposure rationale	Build trust and perception of fairness
Internal documentation	Record rationale for $\phi(g)$ and fairness targets	Ensures institutional memory and auditability	Support governance and legal compliance
Fairness SLOs	Quantified service-level guarantees (e.g., minimum group performance ratio)	Enables measurable commitments	Maintain fairness continuity over time
Launch review process	Mandatory fairness evaluation before deployment	Prevent regressions from product or model updates	Safeguard fairness as a core quality metric
Feedback loop to measurement	Integrate monitoring insights into Stage 1 metrics	Continuous recalibration with new data	Sustainable, adaptive fairness management

We set up monitoring dashboards and alerts specifically for fairness metrics. For instance, we might have a dashboard that tracks  $EP_t(g)$  for each group  $g$  on a daily or weekly basis, along with its target  $\phi(g)$  and perhaps confidence intervals. Statistical change detection methods such as CUSUM (Cumulative Sum Control Charts) or

drift detection tests are employed to flag when a metric shows a sustained shift. If, say, one group's exposure starts dropping week over week beyond normal random variation, an alert would be sent to the relevant engineering and policy teams to investigate. Similarly, calibration metrics by group are tracked; if the error for one group starts increasing after a new feature launch, that might indicate the model is becoming less accurate for that group and requires retraining or adjustment.

Beyond automated monitoring, we incorporate fairness checks into the platform's governance processes. This means that any significant product change (a new algorithm version, a major UI change that affects search, etc.) must go through a fairness evaluation as part of its approval. We institute policy gates such as: no new model is launched unless it meets or exceeds the current fairness metrics on a set of key segments. This is analogous to how one would not launch a model that performs worse on overall accuracy than the incumbent; here we extend that to group fairness. In cases where a slight dip is unavoidable, it should be explicitly signed off by decision makers and accompanied by a mitigation plan. We also require that experiments aiming to measure fairness improvements have sufficient statistical power on segment metrics, to avoid falsely declaring success or failure due to noise.

One important governance artifact is the runbook for fairness incidents. Just as site reliability engineering has incident response plans for outages, we develop playbooks for what to do if an algorithmic bias incident is detected (for example, a sudden drop in outcomes for a protected group). This runbook might include steps like: pause certain automated updates, activate a backup model, inform the trust and safety team to handle incoming complaints, analyze recent changes that could have caused the drift, and communicate to leadership about user impact. Part of this preparation is also deciding on dual-variable freeze procedures. For instance, if the fairness budgets or dual variables (from Stage 5 or Stage 4) start oscillating or diverging due to an unforeseen scenario, the system might temporarily freeze them at a safe value (essentially falling back to a simpler allocation) until engineers can intervene. Having these protocols in place ensures a rapid and organized response to protect users from prolonged unfair treatment.

Transparency and communication form the final piece of the framework. The platform should consider what explanations or disclosures are given to users (both consumers and providers) about how the marketplace operates fairly. This could be through explainability tools or interfaces that allow a provider to inquire, "Why is my product not showing up frequently?" and receive a meaningful answer. While we cannot disclose sensitive details of the algorithm, we might inform them of factors: for example, "The system aims to give opportunities to all sellers; at the moment, other items with slightly higher predicted match likelihoods are being shown more often, but the system is designed to periodically give every listing a chance to be seen." Additionally, messaging can highlight opportunity guarantees: a commitment that if a provider maintains a high quality (as measured by customer satisfaction, etc.), the system will ensure they eventually get visibility. Phrasing these not as strict quotas but as fair opportunity policies is important (Gentzkow, Shapiro, and Sinkinson 2014). It reassures underrepresented groups that there is light at the end of the tunnel without framing it as the platform playing favorites or enforcing equality of outcome regardless of merit.

Internally, transparency involves documenting the rationale for fairness parameters and targets. For instance, why was  $\phi(g)$  set to a certain value for group  $g$ ? These decisions might be informed by legal considerations, user research, or strategic goals. Clear documentation helps in case these choices are challenged or need revisiting.

Stage 8 ensures that the fairness approach is sustainable. Through ongoing measurement, formal checks before launches, prepared responses for issues, and open communication, the platform can maintain the delicate balance between fairness and efficiency. The key artifacts at this stage include defined fairness Service Level Objectives (SLOs) (e.g., "no group's conversion rate will fall below half of the top group's conversion rate for more than a week"), dashboards and alerts to monitor those SLOs, the incident runbook for bias, and user-facing explanation or transparency content. This stage closes the loop, feeding back into Stage 1's measurement as the cycle continues with new data and new iterations of the algorithms.

### **Interfaces Between Offline and Online Phases**

A core principle of our framework is to ensure that guarantees and constraints enforced at offline model training time carry through to the deployed, live online system. Unless one is careful, there is a real possibility that a model that has been trained with fairness constraints will in fact not be constrained in deployment, either because context has changed, users' behaviour has shifted, or due to disparities between an offline simulation and the actual world. For this reason, we define interfaces between the offline and online phases to: (Ashraf et al. 2020)

First, there is a model contract that the training process outputs alongside the model parameters. This contract summarizes the target fairness quantities ( $\phi(g)$  for each group, minimum acceptable values  $b_g$ , the calibration error bounds, etc.) and perhaps the final dual variables or implicit prices of those constraints. When the model is deployed, the online allocation system (Stage 5's mechanisms) reads this contract and configures the initial settings (like the exposure budgets  $B_t(g)$  or the group offsets  $\mu_g$ ) to be consistent with the intended targets. In essence, the offline training says "this model was optimized assuming group A should get at least 10% exposure" and the online system takes that as an input to begin enforcing it from day one.

Second, we warm-start the online fairness controls using the learned state from offline. For example, if the dual variable  $\lambda_A$  for group A was high at the end of training, that suggests the unconstrained model was significantly under-serving A and had to be heavily penalized to meet the constraint. Correspondingly, we might initialize  $\mu_A$  (the online boost for group A) to a relatively large value, rather than starting it at zero and waiting for the online system to slowly increase it. By warm-starting in this way, we avoid a scenario where the first deployment of the model begins by violating fairness goals and only later corrects during that initial period, real users could be negatively impacted. Warm-starting ensures the model's deployment begins in a state as close as possible to fairness compliance.

Third, we implement variance budgeting for exploration. The offline analysis in Stage 7 can tell us how uncertain certain fairness metrics are—for example, how much variability was in  $EP_t(g)$  estimates under different random seeds or splits. We set bounds on the random exploration parameters online—such as the fraction of random impressions, or the distribution of Thompson sampling draws—such that the additional variance introduced will not overwhelm our ability to maintain fairness within acceptable confidence. If our offline replay suggests that to detect a 2% change in  $EP(g)$  we need  $X$  amount of data, we make sure the online randomization does not reduce effective sample size too much for those metrics. In practice, this might mean dynamically scaling down exploration if we detect high variance in outcomes for a group (Patil and Rahman 2022). The exploration policy can have a feedback loop where it monitors group-level statistics: if the results for a minority group are very noisy—for instance, one day they spike, another day they drop—the system might temporarily reduce randomization to stabilize performance for that group before probing further.

Fourth, we perform shadow tests and stress tests of the whole pipeline in a sandbox setting before its complete deployment. In a shadow test, the novel algorithm is run in parallel on live data but does not affect the actual user experience—just to gather data. In such a test, we can intentionally make the distribution of inputs vary to simulate worst-case scenarios for fairness: for instance, we could simulate a surge of traffic from a region that corresponds mostly to one segment of providers or a situation when some group has an unusually high failure rate, just to see whether the system overreacts or handles it gracefully. Such stress tests may show that under extreme but plausible conditions, update rules of the budget or other mechanisms could be unstable. If problems are found—such as oscillating values of  $\mu_g$  or increasing latencies due to too heavy computations of fairness under load—we refine the algorithms or add appropriate safeguards before the real users see the system.

In essence, the interfaces between the offline and online ensure consistency: the fairness intentions set in the lab actually materialize in the field. They also provide a safety net: starting the system in a good state, keeping metrics meaningful by controlling the level of randomness, and testing robustness. This in turn prevents the transition from research to production from accidentally creating new biases or violations.

### **Operationalization and Roles**

Putting an end-to-end fairness framework into place in a real-world marketplace platform is as much an organizational challenge as a technical one. Many different teams and stakeholders must come together since responsibility for algorithmic decisions and their outcomes is distributed. A number of different roles within an organization contribute to the lifecycle framework in various ways. (Kojima and Odahara 2021)

The segment ontology is defined and kept by data and measurement teams, Stage 1, who ensure the right data is being collected on the platform to measure fairness. They set up logging of propensities, build dashboards for exposure and calibration metrics, and manage data pipelines computing the daily or weekly fairness reports. They may also be responsible for curating the dataset in Stage 2, applying reweighting or augmentation, and providing the processed data to modeling teams. That is, they provide the measurement foundation and truth-check the system by monitoring for bias continuously.

Modeling and machine learning research teams focus on Stage 3 and Stage 4, designing representation learning approaches and developing the constrained training algorithms. They may consider adversarial debiasing networks offline, proposing appropriate loss functions or regularizers for fairness, and training candidate models. Their work makes sure that a model can meet the twin goals with no unacceptable loss of accuracy. In turn, the modeling team works with data teams to understand the biases in data, as well as infrastructure teams to ensure any new model architecture or training procedure can be deployed at scale.

Engineering and allocation teams implement the real-time mechanisms of Stage 5. They embed the fairness-aware model in the serving system, writing the logic of exposure budgets, ranking adjustments, and exploration policies. This team is responsible for the performance and reliability of the online system (Bae and Jo 2007). They must ensure that fairness interventions result in no crashes, slow responses, or other degradations. In addition to this, they implement the experiment guardrails (Stage 7) and the monitoring hooks (Stage 8) in the code of the system. This team needs to have a good understanding of both the modeling side and the distributed systems side in order to properly bridge offline intentions with online execution. The economics, product, and policy teams decide what the fairness goals should be in the first place, such as what targets  $\phi(g)$  to set, which segments matter, and how to balance fairness against business metrics. They interpret the fairness utility frontier analysis from Stage 7 to select

operating points that align with the company's values and regulatory requirements. Policy experts ensure the interventions comply with laws; for example, some jurisdictions might regulate what attributes can be used for fairness criteria. Product managers drive the overall effort, ensuring that improvements in fairness translate to good user experiences. They help design the user-facing messaging Stage 8 to effectively communicate the platform's fairness effort to users. Trust and safety, or user experience teams, handle the human side of these algorithmic changes. If there are complaints or confusions by users or providers on how the system is treating them, the trust and safety team steps in. For example, if a small vendor asks "why is my item not appearing?", this team helps provide explanations-with the aid of tools from Stage 8-and gathers feedback (Bandiera, Larcinese, and Rasul 2010). Monitoring qualitative signals such as user sentiment or support ticket trends helps catch fairness issues that quantitative metrics might miss. They make sure that even if the numbers say things are fair, the users feel it is fair. All algorithmic fairness efforts are reviewed by a governance or ethics board. This board would periodically review the metrics; validate major policy decisions, such as increasing or decreasing exposure targets, adding new protected segments to monitor; and oversee the incident response process. They provide accountability and help resolve conflicts such as when engineering wants to loosen a fairness constraint for performance reasons, while policy wants it to be strict-this board would weigh in on that decision. They make sure the outcomes of the Fairness Framework align with the organization's public commitments and values. Fairness is integrated into regular operations by the platform, with these roles clearly delineated. It's not just a one-off project for one team; it is a shared responsibility. Recurring cross-team meetings or stand-ups may be set up to talk about the most recent metrics and anomalies. For example, the data team may mention the drift of some metric, the modeling team may offer a hypothesis- maybe the model is starting to underfit to a group-and the engineering team may check the logs for serving issues, while the product team decides whether or not it is time to activate the contingency plans. Documentation and tooling support this collaborative workflow (Frost et al. 2019). The fairness metrics and their definitions must be well-documented for consistency by all. The runbooks and threshold values are decided beforehand so, if a trigger occurred, say an alert that group Z's conversion rate fell, everyone would know who is responsible for taking which action. In other words, operationalizing the framework means embedding it into the DNA of the organization, treating fairness maintenance in a similar way to how one would go about treating reliability or security: it is an aspect of the quality that is continuously monitored and improved.

### 3. Discussion and Limitations

The proposed framework has its limitation and challenges that might be encountered by implementers.

First, fair decisions rely on having enough and reliable data for all segments of interest. The framework operates under the assumption that with techniques such as reweighting and augmentation (Stage 2) and exploration (Stage 5), we can get enough signal about each group to make a fair decision. In reality, there may be groups that are extremely small or contexts that are very sparse, such that even aggressive exploration cannot eliminate uncertainty. In those cases, fairness constraints can become practically unenforceable or very loose. As an example, if a new demographic group of providers joins a platform, and we have virtually no data on them yet, any exposure constraint for that group would be based on speculation. Heavy exploration to gather data on them might degrade user experience (imagine showing many unknown items to users frequently). Thus, there is a trade-off between how fast we can learn to treat a new group fairly and the cost of that learning in terms of short-term user satisfaction (Teytelboym et al. 2021). Our framework addresses this by allowing soft constraints and gradual pacing, but there remains a judgment call in how to balance inclusion of a new group versus potential immediate losses. Reliable propensity logging (Stage 1) is also assumed; if the logging is flawed, or if the platform initially had a fully deterministic policy (no exploration at all), then the IPS estimates and counterfactual evaluations become tricky, as some items might have zero probability of having been seen (the positivity problem).

Another challenge is that the choice of fairness target  $\phi(g)$  is inherently normative. For example, if we choose to allocate exposure strictly proportional to merit (as estimated by  $qt(s)$ ), we are essentially saying the platform's definition of fairness is "reward true quality regardless of group." But if  $qt(s)$  itself is biased by historical or societal factors-for example, maybe user ratings are more harsh for some minority providers-then merit-proportional exposure can still perpetuate bias. If, on the other hand, we impose an explicit fairness target such as "each group gets equal average exposure", then we may be sacrificing efficiency and possibly even user relevance. The framework allows for any frontier or target to be chosen, but it does not solve what the right fairness definition is for a given context. That requires stakeholder deliberation, legal input, and possibly public consultation. Thus, one must be careful: the framework can enforce a target, but choosing that target is a separate governance decision. The system will only be as just as the targets it is given.

We also have to consider the possibility of gaming and strategic responses: if the platform signals publicly or even implicitly that fairness is being enforced for certain groups, then actors may try to exploit this. For instance, a provider not from a disadvantaged group may try to misclassify themselves as such a group in order to get a boost - this could occur if group membership is self-reported, or inferred indirectly (Eliasz and Spiegler 2016). Currently, the framework assumes groups are known and fixed, but robust governance Stage 8 will have to guard against manipulation of group status. Providers who do get boosts may also come to rely on them and not invest in quality

improvement - a form of moral hazard. The platform should make sure that any fairness interventions are combined with encouragement for all providers to improve their offerings: fairness is about equalizing opportunity to be seen, given quality, not about relieving anyone from striving to satisfy users.

Another limitation is related to perception: even if we can successfully enforce the fairness metrics internally, users may still feel outcomes are unfair. Perhaps the metrics are fair and show each demographic of sellers gets an equal chance, but a certain seller might go several days without sales and feel discriminated against. The personal experience may not be reflective of the aggregate statistics that the platform monitors, which then suggests the need for very good communication-Stage 8's transparency-and possibly micro-level fairness nudges. For example, ensuring that every individual gets some minimum number of impressions over some time might be another layer of fairness not covered by group averages. We did not discuss individual fairness (treating similar individuals similarly) explicitly within this framework, focusing more on group fairness. Sometimes, individual fairness can conflict with group objectives, and it would also further complicate the system. From a technical standpoint, stability is a concern. The multi-stage approach means there are a lot of moving parts (Layton 2018). Dual variables controlling one aspect, pacers controlling another, exploration adding randomness - the interplay can become complex. It is possible for feedback loops to form among these mechanisms too, potentially causing oscillations. For example, the evaluation Stage 7 might drive changes that feed back into the configuration of Stage 5. Tuning all these hyperparameters-lessons learned, thresholds, decay factors-is non-trivial. In practice, extensive simulation and gradual ramp-up in live environments would be required to make sure the system finds a stable equilibrium. We expect that in early iterations, some constraints might be temporarily relaxed if they cause instability, and then gradually tightened as confidence in the system grows. Finally, the long-term simulation that occurs in Stage 7 is only as good as the model of user and provider behavior on which it is based. If our assumptions about how providers respond to more or less exposure are incorrect, then the actual long-term impact could be different from the one predicted. This involves an inherent uncertainty in predicting human behavior within a complex system. For that reason, the platform should consider those simulation results as one input but also base itself on real longitudinal studies once interventions are set in place. Continuous learning is needed; the framework will likely need revisions as new patterns emerge-for instance, the emergence of a new type of provider or changes in user preferences (Casey, Ali-Vehmas, and Valovirta 2017). In summary, our framework provides a structured approach to mitigating bias, but it is by no means plug-and-play. It requires thoughtful implementation, constant monitoring, and adaptation. It requires a very strong organizational commitment toward fairness, even when it may run somewhat against the grain of short-term gains. The benefit we argue is a more sustainable marketplace where trust and diversity are preserved, leading to healthier growth and user loyalty. Reaching that goal will involve working around the limitations described above, areas themselves constituting directions for further research and refinement of this framework.

#### 4. Conclusion

Algorithmic bias in two-sided marketplaces is a complex, dynamically evolving problem. Unlike a static classification task where fairness can be evaluated on a fixed dataset, in a marketplace, interactions are ongoing, there are feedback loops, and biases potentially amplify over time. In this work, we presented a holistic framework that addresses bias at all stages of the marketplace platform lifecycle, from data collection and preparation to model training and real-time decision-making, and finally to monitoring and governance. By treating exposure as a resource to be fairly allocated and incorporating fairness constraints and objectives at multiple stages, this framework is intended to ensure that all participants enjoy a fair opportunity to succeed according to merit.

Our end-to-end approach, from measurement, data debiasing, representation learning, constrained optimization, fair allocation, feedback shaping, to rigorous evaluation and governance, is both comprehensive and practical.

This decomposition allows various teams to take ownership of subproblems but ties them together through shared goals and interface points (Acquisti, Taylor, and Wagman 2016). For instance, the output at training time-the dual variables specifying which groups need boosts-feed directly into serving-time budget adjustments in ranking, linking offline analysis to online behavior. Likewise, the monitoring in production feeds back to data collection and model refinement.

The framework does not eliminate the need for hard choices-such as how to define fairness, and how to trade off fairness against revenue-but it provides a structured way to implement whatever choices the platform decides to make. It emphasizes transparency and gradual change (through pacers and careful experiments) to avoid shocking users with sudden shifts, thereby safeguarding the user experience while moving towards fairer outcomes. By including user perception and trust in the loop-with explainability and surveys-it also recognizes that fairness is partly about what the numbers say and partly about what the stakeholders feel. Future work can extend this framework to incorporate notions of individual fairness, or tackle multi-sided markets with more than two parties, such as advertisers, publishers, and users in an ad network. Another direction is to develop increasingly automated means to tune the many parameters in the system (for example, learning rates for budgets or thresholds for alerts) using techniques from control theory or meta-learning. Fairness begets fairness: the more diverse participation and richer content and competition in a marketplace perceived as fair, the better the overall quality, innovation, and

value to all its users. Getting there is not easy, of course; it takes persistent work and adaptation. Yet the long-term reward is well worth it—a more resilient and trusted platform ecosystem. The framework outlined is a first step toward operationalizing that vision; it offers a life-cycle view of fairness that aligns immediate optimization with the platform’s broader responsibility to its community (Haucap and Heimeshoff 2013) (Padilla, Piccolo, and Vasconcelos 2022) (Querbes 2017).

## References

- [1] Acquisti, Alessandro, Curtis R. Taylor, and Liad Wagman. 2016. “The Economics of Privacy.” *Journal of Economic Literature* 54 (2): 442–92. <https://doi.org/10.1257/jel.54.2.442>.
- [2] Ashraf, Nava, Natalie Bau, Corinne Low, and Kathleen L. McGinn. 2020. “Negotiating a Better Future: How Interpersonal Skills Facilitate Intergenerational Investment.” *The Quarterly Journal of Economics* 135 (2): 1095–1151. <https://doi.org/10.1093/qje/qjz039>.
- [3] Bae, Sung C., and Hoje Jo. 2007. “Underwriter Warrants, Underwriter Reputation, and Growth Signaling.” *Review of Quantitative Finance and Accounting* 29 (2): 129–54. <https://doi.org/10.1007/s11156-007-0030-2>.
- [4] Bandiera, Oriana, Valentino Larcinese, and Imran Rasul. 2010. “Heterogeneous Class Size Effects: New Evidence from a Panel of University Students.” *The Economic Journal* 120 (549): 1365–98. <https://doi.org/10.1111/j.1468-0297.2010.02364.x>.
- [5] Blanchet, Didier, and Marc Fleurbaey. 2020. “Construire Des Indicateurs de La Croissance Inclusive Et de Sa Soutenabilité : Que Peuvent Offrir Les Comptes Nationaux Et Comment Les Compléter ? / Building Indicators for Inclusive Growth and Its Sustainability: What Can the National Accounts Offer and How Can They Be Supplemented?” *Economie Et Statistique / Economics and Statistics* 517 (517): 9–25. <https://doi.org/10.24187/ecostat.2020.517t.2020>.
- [6] Casey, Thomas R., Timo Ali-Vehmas, and Ville Valovirta. 2017. “Evolution Toward an Open Value System for Smart Mobility Services: The Case of Finland.” *Competition and Regulation in Network Industries* 18 (1-2): 44–70. <https://doi.org/10.1177/1783591717734808>.
- [7] Chaudhuri, Ananish. 2016. “Recent Advances in Experimental Studies of Social Dilemma Games.” *Games* 7 (1): 7–7. <https://doi.org/10.3390/g7010007>.
- [8] Chen, Jie, and John A. Rizzo. 2010. “Pricing Dynamics and Product Quality: The Case of Antidepressant Drugs.” *Empirical Economics* 42 (1): 279–300. <https://doi.org/10.1007/s00181-010-0426-z>.
- [9] Choi, Syngjoo, Lars Nesheim, and Imran Rasul. 2015. “Reserve Price Effects in Auctions: Estimates from Multiple Regression-Discontinuity Designs.” *Economic Inquiry* 54 (1): 294–314. <https://doi.org/10.1111/ecin.12226>.
- [10] Cummings, Michael E., Hans Rawhouser, Silvio Vismara, and Erin L. Hamilton. 2019. “An Equity Crowdfunding Research Agenda: Evidence from Stakeholder Participation in the Rulemaking Process.” *Small Business Economics* 54 (4): 907–32. <https://doi.org/10.1007/s11187-018-00134-5>.
- [11] Eliaz, Kfir, and Ran Spiegler. 2016. “Search Design and Broad Matching.” *American Economic Review* 106 (3): 563–86. <https://doi.org/10.1257/aer.20150076>.
- [12] Ellerman, David. 2014. “Parallel Experimentation: A Basic Scheme for Dynamic Efficiency.” *Journal of Bioeconomics* 16 (3): 259–87. <https://doi.org/10.1007/s10818-014-9175-y>.
- [13] Frost, Jon, Leonardo Gambacorta, Yi Huang, Hyun Song Shin, and Pablo Zbinden. 2019. “BigTech and the Changing Structure of Financial Intermediation.” *Economic Policy* 34 (100): 761–99. <https://doi.org/10.1093/epolic/eiaa003>.
- [14] Gamper, Harriet Claire. 2012. “How Can Internet Comparison Sites Work Optimally for Consumers.” *Journal of Consumer Policy* 35 (3): 333–53. <https://doi.org/10.1007/s10603-012-9195-8>.
- [15] Gentzkow, Matthew, Jesse M. Shapiro, and Michael Sinkinson. 2014. “Competition and Ideological Diversity: Historical Evidence from US Newspapers †.” *American Economic Review* 104 (10): 3073–3114. <https://doi.org/10.1257/aer.104.10.3073>.
- [16] Goldfarb, Avi, and Catherine Tucker. 2019. “Digital Economics.” *Journal of Economic Literature* 57 (1): 3–43. <https://doi.org/10.1257/jel.20171452>.
- [17] Hagiu, Andrei, and David B. Yoffie. 2013. “The New Patent Intermediaries: Platforms, Defensive Aggregators and Super-Aggregators.” *Journal of Economic Perspectives* 27 (1): 45–66. <https://doi.org/10.1257/jep.27.1.45>.
- [18] Haucap, Justus, and Ulrich Heimeshoff. 2013. “Google, Facebook, Amazon, eBay: Is the Internet Driving Competition or Market Monopolization?” *International Economics and Economic Policy* 11 (1): 49–61. <https://doi.org/10.1007/s10368-013-0247-6>.
- [19] Havrylychuk, Olena, and Marianne Verdier. 2018. “The Financial Intermediation Role of the P2P Lending Platforms.” *Comparative Economic Studies* 60 (1): 115–30. <https://doi.org/10.1057/s41294-017-0045-1>.
- [20] Jacobides, Michael G., and Ioannis Lianos. 2021. “Ecosystems and Competition Law in Theory and Practice.” *Industrial and Corporate Change* 30 (5): 1199–229. <https://doi.org/10.1093/icc/dtab061>.
- [21] Kojima, Fuhito, and Hiroaki Odahara. 2021. “Toward Market Design in Practice: A Progress Report.” *Japanese Economic Review (Oxford, England)* 73 (3): 1–18. <https://doi.org/10.1007/s42973-021-00103-w>.
- [22] Koutroumpis, Pantelis, Aija Leiponen, and Llewellyn D. W. Thomas. 2020. “Markets for Data.” *Industrial and Corporate Change* 29 (3): 645–60. <https://doi.org/10.1093/icc/dtaa002>.
- [23] Layton, Roslyn. 2018. “Net Neutrality and Mobile App Innovation in Denmark and Netherlands 2010–2016.” *Review of Network Economics* 17 (3): 207–24. <https://doi.org/10.1515/rne-2019-0012>.
- [24] Liu, Xinyuan, Zaiyan Wei, and Mo Xiao. 2019. “Platform Misplicing and Lender Learning in Peer-to-Peer Lending.” *Review of Industrial Organization* 56 (2): 281–314. <https://doi.org/10.1007/s11151-019-09733-2>.
- [25] Ljungqvist, Alexander, Felicia C. Marston, and William J. Wilhelm. 2006. “Competing for Securities Underwriting Mandates: Banking Relationships and Analyst Recommendations.” *The Journal of Finance* 61 (1): 301–40. <https://doi.org/10.1111/j.1540-6261.2006.00837.x>.

- [26] Naudé, Wim. 2022. "Late Industrialisation and Global Value Chains Under Platform Capitalism." *Journal of Industrial and Business Economics* 50 (1): 91–119. <https://doi.org/10.1007/s40812-022-00240-2>.
- [27] Padilla, Jorge, Salvatore Piccolo, and Helder Vasconcelos. 2022. "Business Models, Consumer Data and Privacy in Platform Markets." *Journal of Industrial and Business Economics* 49 (3): 599–634. <https://doi.org/10.1007/s40812-022-00218-0>.
- [28] "PAPERS FROM ACTUARIAL JOURNALS WORLDWIDE." 2016. *Annals of Actuarial Science* 10 (1): 120–68. <https://doi.org/10.1017/s1748499515000147>.
- [29] ———. 2017. *Annals of Actuarial Science* 11 (1): 164–212. <https://doi.org/10.1017/s1748499517000033>.
- [30] Patil, Tejaswi, and Zillur Rahman. 2022. "Mapping the Cause-Related Marketing (CRM) Field: Document Co-Citation and Bibliographic Coupling Approach." *International Review on Public and Nonprofit Marketing* 20 (2): 491–520. <https://doi.org/10.1007/s12208-022-00347-1>.
- [31] Querbes, Adrien. 2017. "Banned from the Sharing Economy: An Agent-Based Model of a Peer-to-Peer Marketplace for Consumer Goods and Services." *Journal of Evolutionary Economics* 28 (3): 633–65. <https://doi.org/10.1007/s00191-017-0548-y>.
- [32] Reurink, Arjan. 2018. "FINANCIAL FRAUD: A LITERATURE REVIEW: FINANCIAL FRAUD: A LITERATURE REVIEW." *Journal of Economic Surveys* 32 (5): 1292–1325. <https://doi.org/10.1111/joes.12294>.
- [33] Ribeiro, Vitor Miguel, and Lei Bao. 2021. "Professionalization of Online Gaming? Theoretical and Empirical Analysis for a Monopoly-Holding Platform." *Journal of Theoretical and Applied Electronic Commerce Research* 16 (4): 682–708. <https://doi.org/10.3390/jtaer16040040>.
- [34] Ribeiro-Navarrete, Samuel, Juan Piñero-Chousa, M. Ángeles López-Cabarcos, and Daniel Palacios-Marqués. 2021. "Crowdlending: Mapping the Core Literature and Research Frontiers." *Review of Managerial Science* 16 (8): 2381–411. <https://doi.org/10.1007/s11846-021-00491-8>.
- [35] Süßmuth, Bernd. 2021. "The Mutual Predictability of Bitcoin and Web Search Dynamics." *Journal of Forecasting* 41 (3): 435–54. <https://doi.org/10.1002/for.2819>.
- [36] Teytelboym, Alexander, Shengwu Li, Scott Duke Kominers, Mohammad Akbarpour, and Piotr Dworczak. 2021. "Discovering Auctions: Contributions of Paul Milgrom and Robert Wilson\*." *The Scandinavian Journal of Economics* 123 (3): 709–50. <https://doi.org/10.1111/sjoe.12441>.
- [37] Ugolini, Marta Maria. 2021. "Leveraging Intersections in Management Theory and Practice (Extended Abstracts)." *Sinergie Italian Journal of Management*, October, 1–624. <https://doi.org/10.7433/srecp.ea.2021.01>.
- [38] "Vol. 3, No. 2 (Full Issue)." 2004. *Journal of Modern Applied Statistical Methods* 3 (2). <https://doi.org/10.22237/jmasm/1099267200>.
- [39] Welfens, Paul J. J., Jens K. Perret, and Deniz Erdem. 2010. "Global Economic Sustainability Indicator: Analysis and Policy Options for the Copenhagen Process." *International Economics and Economic Policy* 7 (2): 153–85. <https://doi.org/10.1007/s10368-010-0165-9>