FairMT: Fairness for Heterogeneous Multi-Task Learning

Guanyu Hu[1,3,4] **Tangzheng Lian**[5] **Na Yan**[6] **Dimitrios Kollias**[1,2*]
Xinyu Yang[3] **Oya Celiktutan**[5] **Siyang Song**[7*] **Zeyu Fu**[7*]

[1]Centre for Multimodal AI, Queen Mary University of London, UK
[2]Digital Environment Research Institute, Queen Mary University of London, UK
[3]School of Computer Science and Technology, Xi'an Jiaotong University, China
[4]Department of Computer Science, Aalto University, Finland
[5]King's College London, London, UK
[6]Northeastern University, China
[7]University of Exeter, UK

Abstract

Fairness and predictive utility are assessed at individual outputs but shaped by shared updates in heterogeneous multi-task learning. Multi-class classification, binary detection, and scalar regression draw on common features from one encoder, allowing updates from one task to affect predictions and group disparities in the others. Utility-oriented optimizers balance aggregate losses, but those losses can conceal group-specific errors. Single-output fairness criteria characterize these output-specific disparities, but their raw scales give no common basis for setting correction priorities. The task heads can also turn comparable corrections into different encoder updates, with different implications for predictive utility. Joint training therefore calls for prioritizing disparities across outputs and accounting for how their correction affects predictive utility. We propose FairMT, which frames fair heterogeneous multi-task learning through a coordinate-allocation view. Group shortfalls relative to the best-performing group on each measure become coordinates of a shared fairness constraint, which assigns greater corrective pressure to larger violations on a calibrated scale. FairMT's head-aware proxy estimates the alignment among utility gradients and their interaction with the allocated fairness correction through the task heads, making the task weights responsive to the current correction. On vision and language benchmarks, FairMT reduces group disparities for all three output types with utility close to that of utility-oriented optimizers.

1 Introduction

Fair multi-task learning is important for combining advances in predictive modeling [26, 24, 55, 54, 53, 22, 18] with equitable performance across demographic groups [23, 33]. Algorithmic fairness is typically built on a single-predictor abstraction [41, 32] (one model, one output space, one fairness notion, and one set of group-level constraints). This abstraction is misaligned with heterogeneous multi-task learning (MTL) [7, 46, 61], where multi-class classification, binary detection, and scalar regression can exploit common structure through a shared encoder [27]. Their distinct output spaces can call for different fairness criteria and group-level constraints. Updating the encoder to reduce missed detections for one group can also change class probabilities and regression errors. Fair joint

Email: guanyu.hu@aalto.fi. *Corresponding authors: Dimitrios Kollias (d.kollias@qmul.ac.uk), Siyang Song (s.song@exeter.ac.uk), and Zeyu Fu (z.fu@exeter.ac.uk).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

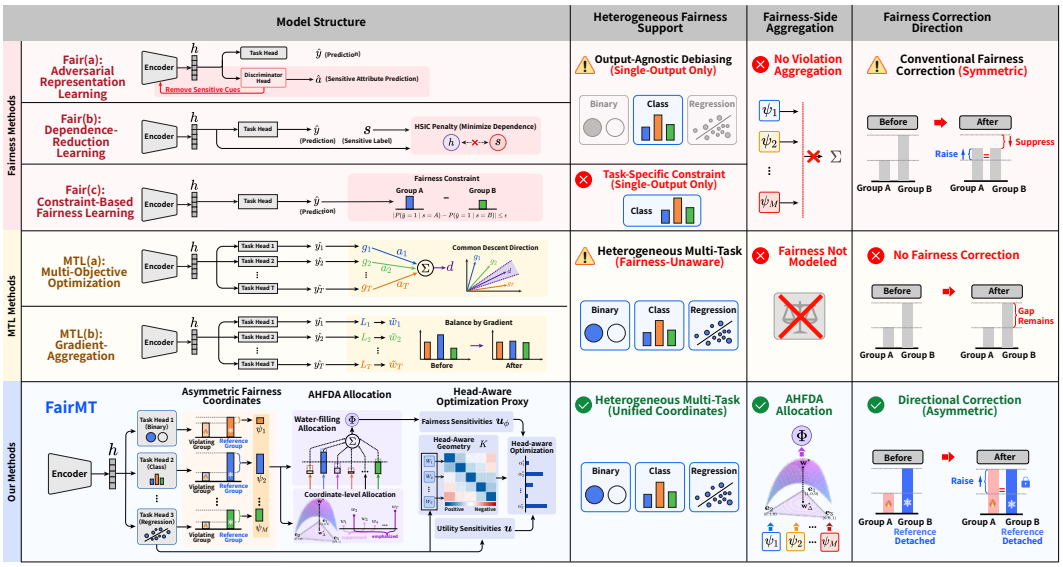

Figure 1: **Conceptual comparison of fairness, MTL, and FairMT paradigms.** Fairness methods debias representations or constrain individual outputs (Fair(a–c)), while utility-oriented MTL methods coordinate predictive objectives across tasks (MTL(a–b)). FairMT directs correction toward group shortfalls using asymmetric coordinates. AHFDA combines violations in a shared constraint that gives more weight to larger gaps. The head-aware proxy weights tasks using head-space estimates of fairness–utility gradient alignment.

training therefore requires a common basis for comparing disparities across outputs and must account for how corrections at individual outputs affect predictive utility across tasks.

Figure 1 illustrates why existing fairness and MTL paradigms do not directly yield fair heterogeneous MTL. On the fairness side, adversarial methods such as LNL [29] suppress sensitive information in learned features [39] (Fair(a)) and UFaTE-style objectives [12] penalize dependence on sensitive attributes (Fair(b)), both leaving open which group's predictions need correction and risking the loss of information needed for prediction [33]. Constraint-based methods (Fair(c)) such as Equalized Odds [19], covariance-style constraints [59, 58], and reductions-based fair classification [2] target group disparities within individual tasks, as do fairness regularization [44, 6] and group reweighting [40], leaving cross-output comparisons unresolved. Combining these disparities in a shared constraint exposes ***(1) Aggregation-Incompatible Heterogeneity***. Fairness metrics are meaningful in their native output spaces, but rate-based detection gaps, class-wise one-vs-rest disparities, and regression-error gaps are not directly commensurable. In a shared constraint, numerical scale can therefore give one disparity disproportionate influence. Even when gaps are comparable, a symmetric penalty leaves the direction of correction open, giving rise to ***(2) Non-Directional Correction***. The penalty depends on the size of the gap, which can shrink either when predictions improve for the worse-performing group or when they deteriorate for the better-performing group.

On the MTL side, utility-oriented optimizers coordinate competing predictive objectives in a shared update. MGDA [48] seeks a common descent direction (MTL(a)), while GradNorm [8] adapts task-loss weights and PCGrad [57] projects conflicting gradients (MTL(b)). Group-wise evaluations reveal disparities that aggregate task losses can conceal [21, 23, 51, 52]. Even minimizing the largest group-level prediction loss within a task [47] leaves open how to combine comparable, directional fairness violations across outputs in a shared constraint, giving rise to ***(3) Missing Fairness Aggregation Proxy***. Averaging violations uniformly can dilute a severe disparity, while a hard maximum can switch the correction abruptly as estimates fluctuate. Coordinating utility gradients also leaves open how the aggregated fairness signal reaches the shared encoder through the task heads, exposing ***(4) Head-Dependent Fairness Propagation***. Head scale, sensitivity, and decision geometry can make comparable corrections produce different encoder updates and affect predictive losses differently.

These challenges motivate **FairMT**, which frames fair heterogeneous MTL through a coordinate-allocation view. To address ***(1)***, the *Fairness-Coordinate Interface* represents disparities measured in each task's native output space as violation coordinates. A multi-class task, for example, has one predictive objective but can have a separate fairness coordinate for each class. Validation-calibrated scales make these violations comparable across outputs. To address ***(2)***, the *Asymmetric*

Fairness Coordinates use each statistic's best-performing valid group as a reference held fixed during differentiation, directing the explicit fairness gradient toward the largest group shortfalls. To address ***(3)***, *Asymmetric Heterogeneous Fairness Disparity Aggregation* (AHFDA) combines these normalized, directional violations into one differentiable constraint, giving larger violations more weight while allowing several severe gaps to contribute. To address ***(4)***, the *Head-Aware Optimization Proxy* estimates in head space how utility gradients align with one another and with the resulting fairness gradient, then uses these alignments to set task weights for the shared update.

Across vision and language benchmarks, FairMT reduces group disparities for all three output types. Its predictive utility remains close to that of utility-oriented MTL optimizers.

2 Problem Setup

Heterogeneous MTL setup. Let $\mathcal{T} = \{1, \dots, T\}$ index the utility tasks, spanning binary detection, multi-class classification, and scalar regression. Each sample pairs an input x_i with group metadata a_i. Datasets may annotate different subsets of tasks [25]. The binary observation mask $m_{i,t}$ indicates which annotations are available, taking value one when the label or target $y_{i,t}$ is observed.

A shared encoder f_{θ_s} supplies features to a separate head for each task, yielding the prediction $\hat{y}_{i,t} = h^{(t)}_{\theta_t}(f_{\theta_s}(x_i))$ for task t. We collect the encoder and head parameters in θ. With the observation mask, the empirical risk for a decomposable utility loss on mini-batch $\mathcal{B}$ is

$$\widehat{L}^{\mathcal{B}}_t(\theta) = \frac{\sum_{i \in \mathcal{B}} m_{i,t} \ell_t(\hat{y}_{i,t}, y_{i,t})}{\max(1, \sum_{i \in \mathcal{B}} m_{i,t})}, \tag{1}$$

where ℓ_t is the sample-wise utility loss for task t. Tasks without labels or targets in the batch receive zero utility weight. Multi-class classification utility uses standard softmax cross-entropy. For regression, Eq. (1) gives the masked mean squared error (MSE), while the concordance correlation coefficient (CCC) uses all observed targets and their predictions in the batch (Appendix F.4).

Output-side fairness coordinates. For each fairness setting, FairMT uses a fixed map π to assign sample i to group $s_i = \pi(a_i)$ in the group set $\mathcal{G}$, such as Gender, Race, or Age×Gender groups. A fairness coordinate specifies the output at which these groups are compared. Binary detection contributes one coordinate per task, multi-class classification contributes one per class through a one-vs-rest view, and scalar regression contributes one per target. The M coordinates form $\mathcal{Q}$, partitioned by output type into $\mathcal{Q}_{\text{bin}}$, $\mathcal{Q}_{\text{cls}}$, and $\mathcal{Q}_{\text{reg}}$. Each coordinate q inherits the observation mask of its parent task $t(q)$ through $m_{i,q} = m_{i,t(q)}$. The same mask determines which predictions enter the utility loss and the corresponding group comparisons.

Fairness-violation interface. Each coordinate q measures group disparity in its native output space, yielding a nonnegative violation $\psi_q(\theta)$. FairMT collects these raw violations in $\Psi(\theta)$. To compare violations across outputs, FairMT divides each entry by a fixed scale $\nu_q > 0$ calibrated on validation data. The normalized vector $\bar{\Psi}(\theta)$ has entries $\bar{\psi}_q(\theta) = \psi_q(\theta)/\nu_q$. In each mini-batch, AHFDA aggregates the normalized violations from coordinates with valid group comparisons.[1]

Fairness-constrained MTL objective. Let $L_t(\theta)$ denote the population risk for task t. The mini-batch losses supply the utility gradients during training. The population formulation collects the task risks into a vector objective under a single fairness constraint. The shifted AHFDA aggregator $\widetilde{\Phi}_\mu$ (Section 3.2) combines the normalized violations. With $\mu > 0$ controlling allocation sharpness and $\epsilon \geq 0$ bounding the aggregate violation, the joint learning problem is

$$\min_{\theta} \big(L_1(\theta), \dots, L_T(\theta)\big) \quad \text{s.t.} \quad \widetilde{\Phi}_\mu(\bar{\Psi}(\theta)) \leq \epsilon. \tag{2}$$

The task risks and the fairness constraint depend on the same encoder parameters. Section 3.3 uses the first-order conditions for constrained Pareto stationarity to derive a shared update direction that balances the task gradients against the aggregate fairness gradient.

[1] Appendix A specifies the criteria for valid group comparisons, the calibration of coordinate scales, and the AHFDA objective on active coordinates.

3 FairMT Framework

This section introduces the proposed **FairMT** framework. Training under Eq. (2) requires deciding which group shortfalls to correct and how those corrections should shape the shared update. *Asymmetric Fairness Coordinates* measure shortfalls relative to the best-performing group on each statistic. *AHFDA* combines the normalized violations into a single constraint that prioritizes larger relative shortfalls. Task heads can turn comparable corrections into different encoder updates. The *Head-Aware Optimization Proxy* estimates how utility and fairness gradients align through these heads and uses that estimate to weight the task losses in the primal–dual training loop.

3.1 Asymmetric Fairness Coordinates

Asymmetric Fairness Coordinates instantiate the violation interface in Section 2. Each coordinate retains the group statistics suited to its output and measures shortfalls against the best-performing valid group on each statistic. In the forward pass, each valid discrete component and each valid regression coordinate equals the squared largest pairwise gap in its group statistic. FairMT recomputes the references in each mini-batch and holds them fixed during differentiation via the stop-gradient operator $\mathrm{sg}(\cdot)$. The explicit fairness gradient for each component then acts through the group statistics with the largest positive shortfalls.[2]

Discrete coordinates. Binary detection and one-vs-rest multi-class classification share the same group-rate construction. Within each group, FairMT averages predicted positive scores over observed positives for the soft true-positive rate (TPR) and over observed negatives for the soft false-positive rate (FPR). The indicators I_q^+ and I_q^- mark whether at least two groups have observed positives or negatives, respectively. For valid comparisons, A_q^{TPR} measures the largest squared shortfall below the highest valid TPR, and A_q^{FPR} measures the largest squared excess above the lowest valid FPR. FairMT detaches each available reference and sets unavailable components to zero. With $\beta \geq 0$ weighting the FPR component, the raw discrete violation is

$$\psi_q^{\mathrm{disc}}(\theta) = \frac{I_q^+ A_q^{\mathrm{TPR}} + \beta I_q^- A_q^{\mathrm{FPR}}}{I_q^+ + \beta I_q^-}, \tag{3}$$

with the coordinate omitted when $I_q^+ + \beta I_q^- = 0$. The denominator keeps a missing component from diluting the other. Setting $\beta = 0$ yields a TPR-only equal-opportunity variant, while $\beta = 1$ yields an equalized-odds variant with equal weights for the available rates. When score corrections reduce false negatives or false positives in lagging groups, parity and accuracy can improve.

Regression coordinates. For a scalar regression target, FairMT compares group prediction errors. The group error $\widehat{e}_{g,q}$ averages the prediction error over observed targets in group g, using mean squared error on normalized targets in our experiments. When at least two groups have observed targets, FairMT uses the lowest group error as the detached reference, $e_q^{\mathrm{ref}} = \mathrm{sg}(\min_g \widehat{e}_{g,q})$. The raw regression violation measures the largest squared excess above this reference,

$$\psi_q^{\mathrm{reg}}(\theta) = \max_g \left[\widehat{e}_{g,q} - e_q^{\mathrm{ref}}\right]_+^2, \tag{4}$$

where $[x]_+ = \max(x, 0)$ and both extrema range over groups with observed targets. Lower errors in lagging groups can improve error parity and prediction quality. We audit regression fairness with Kolmogorov–Smirnov (KS), Coverage Shift Probability (CSP), and binned equalized-odds-style disparities, defined in Appendix F.2. Appendix I.2 connects the surrogate to these audits by bounding the disparities in terms of group-wise residual errors under balance and regularity conditions.

3.2 AHFDA: Adaptive Global Fairness Allocation

The asymmetric coordinates in Section 3.1 produce raw violations $\Psi(\theta)$, which are normalized into comparable coordinates $\bar{\Psi}(\theta)$ as described in Section 2. The constraint then needs to determine how strongly each violation influences the shared update. Uniform averaging can dilute a severe violation, while a hard maximum can shift all fairness weight to a different coordinate as the largest estimate

[2] Appendix B details the mini-batch estimators, component validity and normalization, and the forward gap equivalence. Appendix I.1 derives the corresponding gradient directions.

changes. AHFDA gives priority to larger relative violations while allowing several severe coordinates to contribute to the same constraint when their violations are close in magnitude.

Asymmetric Heterogeneous Fairness Disparity Aggregation (AHFDA) assigns nonnegative weights w_q to the M normalized violations. The weights sum to one, placing the allocation w on the coordinate simplex Δ_M. With regularization strength $\mu > 0$, the aggregate is

$$\Phi_\mu(\bar{\Psi}) = \max_{w \in \Delta_M} \left[w^\top \bar{\Psi} - \frac{\mu}{2} \|w\|_2^2 \right], \qquad \Delta_M = \{w \in \mathbb{R}^M_{\geq 0} : \mathbf{1}^\top w = 1\}, \tag{5}$$

where $\mathbf{1}$ denotes the all-ones vector. The linear term rewards assigning weight to larger violations, and the quadratic penalty makes concentrated allocations costly. At zero violations, the uniform allocation gives $\Phi_\mu(0) = -\mu/(2M)$. We shift this baseline to zero,

$$\widetilde{\Phi}_\mu(\bar{\Psi}) = \Phi_\mu(\bar{\Psi}) + \frac{\mu}{2M}, \tag{6}$$

and impose $\widetilde{\Phi}_\mu(\bar{\Psi}(\theta)) \leq \epsilon$ as the fairness constraint in Eq. (2).[3]

Closed-form water-filling allocation. Removing the quadratic term in Eq. (5) gives the hard maximum $\max_{w \in \Delta_M} w^\top \bar{\Psi} = \max_q \bar{\psi}_q$, with multiple optimal allocations at ties. The penalty makes the objective strongly concave in w, yielding a unique allocation. By Danskin's theorem, this allocation is the gradient of the aggregate with respect to $\bar{\Psi}$ [11].

Theorem 1 (**AHFDA water-filling allocation**). For any $\bar{\Psi} \in \mathbb{R}^M_{\geq 0}$ and $\mu > 0$, Eq. (5) has a unique maximizer, and the shifted aggregate is differentiable. The allocation and gradient satisfy

$$w^\star(\bar{\Psi}) = \Pi_{\Delta_M}\left(\frac{\bar{\Psi}}{\mu}\right), \qquad w_q^\star = \left[\frac{\bar{\psi}_q}{\mu} - \kappa\right]_+, \qquad \nabla_{\bar{\Psi}} \widetilde{\Phi}_\mu(\bar{\Psi}) = w^\star(\bar{\Psi}),$$

where Π_{Δ_M} is Euclidean projection onto the simplex and $\kappa \in \mathbb{R}$ satisfies $\sum_{q=1}^M w_q^\star = 1$.

The simplex projection adapts the threshold κ to the current violations [49]. Only coordinates with $\bar{\psi}_q > \mu\kappa$ receive positive weight, and $w_q^\star$ gives the sensitivity of the constraint to the corresponding normalized violation. Since Euclidean projection is nonexpansive, the exact allocation varies continuously with the violations on a fixed coordinate set. Appendix C.1 gives the proof.

Projection-free training solver. To approximate the allocation in Theorem 1 during training, FairMT applies a short Frank–Wolfe solve to the equivalent simplex quadratic on the active normalized violations [28]. The solver starts at a simplex vertex. Each step moves toward the vertex associated with the largest adjusted violation $\bar{\psi}_q - \mu w_q$, preserving feasibility without projection and adding at most one coordinate to the support. Evaluating the shifted objective at the returned weights gives a lower estimate of the exact aggregate. FairMT holds these weights fixed during backpropagation, using them to approximate the exact gradient with respect to the normalized violations. Appendix C.2 gives the solver, exact line search, and bounds on both approximation errors.

3.3 Head-Aware Optimization Proxy

The shifted AHFDA aggregate in Eq. (6) determines how much each normalized violation contributes to the fairness gradient. This gradient reaches the shared encoder through the task heads, where it interacts with the utility gradients. The *Head-Aware Optimization Proxy* estimates these interactions to choose the task weights for the primal update in Section 3.4.

Ideal constrained direction. The task risks and the fairness constraint in Eq. (2) depend on the same encoder parameters θ_s. With the group references held fixed, the shared-parameter block of the local surrogate's Pareto stationarity condition requires a weighted sum of these gradients to vanish [1, 17].

The task weights α lie on the simplex $\Delta_T = \{\alpha \in \mathbb{R}^T_{\geq 0} : \mathbf{1}^\top \alpha = 1\}$, giving a convex mixture of utility gradients. At iteration r, let $g_t^s = \nabla_{\theta_s} L_t(\theta^{(r)})$ denote the shared gradient of task t and $g_\Phi^s = \nabla_{\theta_s} \widetilde{\Phi}_\mu(\bar{\Psi}(\theta^{(r)}))$ the shared fairness gradient. The current dual multiplier $\lambda^{(r)} \geq 0$ scales the

[3] In a mini-batch, $\bar{\Psi}_\mathcal{B}$ collects the normalized violations on the $M_\mathcal{B}$ active coordinates. AHFDA uses $M_\mathcal{B}$ in place of M. With no active coordinate, the update uses only utility gradients and carries the dual multiplier forward. Appendix C.1 gives the active-set formulation and proof.

fairness gradient and stays fixed in the ideal direction-finding problem

$$\min_{\alpha \in \Delta_T} \frac{1}{2} \left\| \sum_{t=1}^{T} \alpha_t g_t^s + \lambda^{(r)} g_\Phi^s \right\|_2^2 . \tag{7}$$

The residual measures how closely the utility mixture can offset the scaled fairness gradient.[4]

Head-aware fairness routing. Evaluating Eq. (7) directly requires propagating the gradients through the shared encoder. The proxy compares the gradients where the task heads meet the shared representation of dimension D. Each head's derivatives determine how its output-level signals reach this representation. FairMT sums the resulting per-sample representation gradients into $a_t \in \mathbb{R}^D$ for task t's utility loss and $c_t \in \mathbb{R}^D$ for its contribution to the fairness gradient. The fairness direction incorporates AHFDA's coordinate weights and the inverse calibration scales. Each sample contributes one sensitivity-weighted direction for a scalar head or one per logit for a multi-class head.

The head-space residual omits the sample-specific encoder Jacobians that enter the full gradients in Eq. (7). We stack the utility directions as rows of $A \in \mathbb{R}^{T \times D}$ and the fairness directions as rows of C of the same shape. The head-space residual combines the utility mixture $A^\top \alpha$ with $\lambda^{(r)} C^\top \mathbf{1}$, where $C^\top \mathbf{1}$ sums the fairness directions over tasks. Expanding the squared norm and dropping the term independent of α gives the convex task-space quadratic program (QP)[5]

$$\alpha^{(r)} \in \arg\min_{\alpha \in \Delta_T} \frac{1}{2} \alpha^\top K \alpha + b^\top \alpha, \qquad K = AA^\top, \qquad b = \lambda^{(r)} AC^\top \mathbf{1}. \tag{8}$$

A fairness correction that helps one task can affect the utility of other tasks through the shared encoder. In Eq. (8), K captures alignment among utility directions, and b captures their alignment with the aggregate fairness direction $C^\top \mathbf{1}$, scaled by $\lambda^{(r)}$. When $\lambda^{(r)} = 0$ or the mini-batch has no active fairness coordinates, $b = 0$ and the proxy minimizes the norm of the utility mixture in head space. This recovers the MGDA-style criterion when the head-space utility directions coincide with the shared-parameter utility gradients [13, 48].

3.4 FairMT Primal–Dual Optimization

FairMT connects the three modules through a primal–dual loop (Algorithm 1).[6] AHFDA allocates pressure over the active fairness coordinates, and the head-aware proxy combines the resulting fairness direction with the utility directions to weight the active tasks. With both sets of weights and the group references fixed, the primal step updates the encoder and heads. The dual step raises the multiplier for a positive pre-update constraint residual and moves it toward zero for a negative one. Without active fairness coordinates, FairMT uses the utility-only mixture and holds the multiplier fixed. After training, prediction uses only the shared encoder and task heads.

Algorithm 1 FairMT Primal–Dual Optimization

Require: Initial parameters $\theta^{(0)}$, multiplier $\lambda^{(0)} \geq 0$, AHFDA parameter μ, tolerance ϵ, and learning rates $\rho_s, \rho_t, \rho_\lambda$ for the encoder, heads, and multiplier

for each iteration r **do**
  Sample mini-batch $\mathcal{B}$ and identify its active tasks $\mathcal{T}_\mathcal{B}$ and fairness coordinates.
  Compute masked losses and active normalized violations with detached group references.
  if the batch has active fairness coordinates **then**
    AHFDA: compute and detach $w^{(r)}$, form the local aggregate $\widetilde{\Phi}_\mu^{(r)}$, and store $r_\Phi^{(r)} = \widetilde{\Phi}_\mu^{(r)}(\theta^{(r)}) - \epsilon$.
  else
    Set $b = 0$ in Eq. (8).
  end if
  Head-aware proxy: run the solver for Eq. (8) on $\mathcal{T}_\mathcal{B}$ and detach the returned weights $\alpha^{(r)}$.
  Primal update: set $\mathcal{J}^{(r)}(\theta) = \sum_{t \in \mathcal{T}_\mathcal{B}} \alpha_t^{(r)} \widehat{L}_t^\mathcal{B}(\theta)$ and add $\lambda^{(r)}(\widetilde{\Phi}_\mu^{(r)}(\theta) - \epsilon)$ if a fairness coordinate is active.
  Update the encoder and heads using $\nabla_\theta \mathcal{J}^{(r)}(\theta^{(r)})$ with rates ρ_s and ρ_t.
  Dual update: if a fairness coordinate is active, set $\lambda^{(r+1)} = [\lambda^{(r)} + \rho_\lambda r_\Phi^{(r)}]_+$. Otherwise, carry forward $\lambda^{(r)}$.
end for

[4] With $\lambda^{(r)} = 0$, Eq. (7) gives the MGDA-style minimum-norm problem in shared-parameter space. Appendix E gives the stationarity conditions for the detached-reference surrogate and the training procedure.

[5] Training restricts the QP to tasks with observed labels or targets. Appendix D derives the head directions and proxy, and describes the projection-free solver and the MGDA-style special case.

[6] Algorithm 3 in Appendix E details the mini-batch procedure, including masks, active sets, and detached quantities.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate FairMT on three MTL benchmarks that vary in output type, supervision, and input modality (Table 1). **Attribute Detection** uses CelebA [37] with 29 fully supervised binary facial attribute tasks. **Affective Analysis** combines AffectNet [42], AffectNet-VA, and EmotioNet [15] for expression classification, valence–arousal regression, and action-unit (AU) detection. The datasets label different outputs, yielding partial supervision. **Toxicity and Social Bias Analysis** uses Civil Comments from WILDS [30], with toxicity-related annotations also studied in unintended-bias evaluation [5]. Each fairness evaluation uses one fixed group map $s_i = \pi(a_i)$ across all tasks and coordinates. Appendix F details the experimental protocol.

Table 1: Benchmark tasks, supervision, and sensitive groupings. Each multi-class prediction head represents one utility task and contributes one-vs-rest fairness coordinates, one per class.

Benchmark	Datasets	Supervision	# Utility Tasks / # Fairness Coordinates	Outputs and Sensitive Groupings
Attribute Detection	CelebA	Full	29 / 29	29 binary facial attributes; Age and Gender groupings
Affective Analysis	AffectNet, AffectNet-VA, EmotioNet	Partial	11 / 17	Expression classification, valence–arousal regression, AU detection; Age, Gender, and Race groupings
Toxicity & Social Bias Analysis	Civil Comments	Full	10 / 10	9 binary toxicity-related tasks and toxicity regression; Sexual Orientation, Race, and Religion groupings

Evaluation metrics. In the main benchmark comparisons, we evaluate utility with Accuracy for discrete outputs and CCC for regression. For binary detection and one-vs-rest multi-class classification coordinates, we measure disparities with the equal-opportunity gap (EO) and equalized odds difference (EOD). For regression, we audit disparities with KS, CSP, and a binned equalized-odds-style metric (denoted EOD). Appendix F.2 defines each metric and explains aggregation across fairness coordinates.

Baselines. The main comparisons cover MGDA [13, 48] and GradNorm [8] for utility-oriented MTL, LNL [29] and UFaTE [12] for fairness-oriented learning, and MGDA-Mean and MGDA-UFaTE as hybrid baselines. MGDA-Mean adds a fairness constraint to MGDA by uniformly averaging FairMT's active, normalized asymmetric coordinates. Within each benchmark, methods in the main comparisons share data splits, encoders, task heads, and training budgets. We use ConvNeXt-Base for vision and DeBERTa-v3-base for text (Appendices F.3, F.5, and F.6). On CelebA with Gender grouping, we additionally evaluate the native fair-MTL method L2T-FMT [45] and our task-wise FairGrad-MT and GroupDRO-MT adaptations (Appendix K.7).

4.2 Fairness Gains and Predictive Utility

Overall comparison. FairMT achieves the lowest disparity in 39 of the 40 comparisons in Table 2. It matches or exceeds the strongest baseline utility in 13 of 17 settings defined by output type and sensitive grouping, with a maximum shortfall of 0.9 points. MGDA and GradNorm generally preserve utility while leaving larger disparities, and LNL and UFaTE often narrow those gaps at a utility cost. FairMT improves every metric relative to both hybrid baselines, including MGDA-Mean, which uses the same normalized fairness coordinates with uniform aggregation. Appendix G provides detailed benchmark tables and repeated-run results.

Visual and text benchmarks. On fully supervised CelebA, FairMT attains the highest Accuracy and the lowest EOD and EO for both groupings (Table 2(a)). Affective Analysis tests whether those gains extend to a shared encoder serving expression classes, binary AU labels, and continuous valence and arousal targets under partial supervision. Under Age grouping, FairMT more than halves expression and AU detection EOD relative to MGDA while improving Accuracy on both (Table 2(b)). Regression CCC remains comparable to MGDA's, with smaller disparities in prediction distributions (KS) and threshold coverage (CSP). UFaTE leads on binned regression EOD for Age, the sole exception to FairMT's disparity lead. On Civil Comments, where toxicity detection and regression share a text encoder, FairMT reduces Race-group detection EOD by more than half and nearly halves binned regression EOD relative to MGDA while slightly improving both utility measures (Table 2(c)).

Additional comparisons. The disparity reductions extend beyond the default encoders and joint training. With RoBERTa-base on Civil Comments and DINOv2 ViT-B/14 on Affective Analysis, FairMT lowers disparities with comparable utility relative to matched MGDA runs (Appendix K.4). On one task per benchmark, FairMT applies the same fairness objective and leads in utility and every fairness metric (Appendix K.1). The direct CelebA comparison under Gender grouping tests FairMT against other ways of enforcing fairness across tasks. FairMT improves both utility and fairness over L2T-FMT's task-wise objective selection and over the group reweighting used by FairGrad-MT and GroupDRO-MT (Table 3).

4.3 Mechanisms and Trade-offs

The main comparisons establish that lower disparities can coexist with strong utility across output types. The ablations examine how the fairness constraint and the head-aware proxy contribute to this balance. We then vary the training settings and evaluate the trade-offs across tasks and group definitions (Appendices H and I).

Component effects. Holding MGDA's utility weighting rule fixed on CelebA (Gender), adding the AHFDA constraint lowers mean EOD from 22.5 to 8.2 while mean Accuracy remains approximately 87.7% (Table 8). The constraint gives larger normalized violations more influence on the shared update, even for tasks with modest utility weights. Removing AHFDA from full FairMT on Civil Comments (Race) likewise sharply increases detection EOD.

Keeping AHFDA active on CelebA, adding the head-aware proxy raises Accuracy from 86.0% to 88.1% and reduces EOD and EO (Table 8). A fairness correction for one output can also affect other outputs through the shared encoder. The proxy uses head-space alignment to adjust the utility weights in light of the current fairness direction, so the utility mixture can respond to the correction (Section 3.3). On Civil Comments, removing the proxy lowers regression CCC while the fairness constraint remains active.

Table 2: Fairness and utility across benchmarks. Arrows show preferred directions. EO is equal opportunity. EOD is equalized odds difference (binned for regression).

(a) Attribute Detection (in %)

CelebA Binary Detection

Method	Acc ↑	Age		Acc ↑	Gender	
		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA†	87.7	11.4	10.4	87.7	22.5	22.3
GradNorm†	84.8	9.4	8.4	84.8	19.4	18.3
LNL*	80.8	7.8	7.5	79.7	13.6	10.6
UFaTE*	76.4	6.8	5.5	75.4	12.2	10.2
MGDA-Mean†*	76.9	7.7	6.3	74.6	17.8	17.2
MGDA-UFaTE†*	78.6	6.8	4.4	77.8	12.6	9.6
FairMT	**88.4**	**3.4**	**2.0**	**88.1**	**6.4**	**3.9**

(b) Affective Analysis (in %)

AffectNet Expression Classification

Method	Acc ↑	Age		Acc ↑	Gender		Acc ↑	Race	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA†	59.5	23.4	23.4	**59.5**	8.6	8.6	**59.5**	13.9	13.9
GradNorm†	57.5	20.3	19.6	57.5	9.1	9.1	57.5	14.9	14.8
LNL*	53.6	15.1	15.9	55.4	7.1	7.0	53.6	10.9	9.9
UFaTE*	51.1	12.8	12.4	53.4	6.0	6.0	54.0	8.7	8.5
MGDA-Mean†*	51.4	19.9	18.4	56.7	7.2	6.0	55.8	8.4	7.4
MGDA-UFaTE†*	55.6	15.0	13.2	56.4	6.7	6.8	56.2	7.4	7.4
FairMT	**60.6**	**9.9**	**8.3**	**59.5**	**4.0**	**3.9**	59.4	**5.6**	**4.8**

EmotioNet AU Detection

Method	Acc ↑	Age		Acc ↑	Gender		Acc ↑	Race	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA†	81.3	19.1	19.4	81.3	6.5	5.9	81.3	18.6	18.4
GradNorm†	78.4	21.4	20.4	78.4	5.8	5.6	78.4	18.5	18.4
LNL*	69.5	12.9	11.1	73.5	4.5	4.3	74.8	15.4	15.1
UFaTE*	67.3	11.1	10.9	77.4	3.3	4.8	77.8	14.3	13.3
MGDA-Mean†*	62.0	16.5	15.1	74.5	3.7	3.6	72.8	13.4	13.1
MGDA-UFaTE†*	69.6	12.5	11.2	75.4	3.9	3.5	74.7	14.9	13.7
FairMT	**81.6**	**8.8**	**7.5**	**81.8**	**2.3**	**1.8**	**81.8**	**8.1**	**6.8**

AffectNet-VA Regression

Method	CCC ↑	Age			CCC ↑	Gender			CCC ↑	Race		
		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓
MGDA†	**69.3**	19.1	14.0	13.0	69.3	16.5	7.4	5.5	69.2	17.1	13.3	15.7
GradNorm†	67.2	17.8	14.4	13.1	67.5	17.7	7.1	6.7	67.3	18.2	13.0	14.1
LNL*	60.9	16.3	12.3	11.3	63.6	14.9	6.9	5.1	64.3	12.6	8.1	9.6
UFaTE*	58.5	14.3	12.9	**8.6**	63.0	13.1	6.1	5.3	63.8	13.1	7.9	10.9
MGDA-Mean†*	59.0	18.5	13.8	11.5	64.1	14.9	5.5	4.7	62.1	16.6	11.9	13.6
MGDA-UFaTE†*	61.9	17.8	13.4	12.1	65.9	13.8	6.8	5.8	64.3	10.0	7.1	8.7
FairMT	68.9	**13.9**	**7.1**	9.6	**69.6**	**10.7**	**3.7**	**2.2**	**69.7**	**6.2**	**3.3**	**4.9**

(c) Toxicity & Social Bias Analysis (in %)

Civil Comments Detection

Method	Acc ↑	LGBTQ+		Acc ↑	Race		Acc ↑	Religion	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA†	87.1	14.3	14.1	81.6	22.9	22.2	**84.1**	28.1	27.9
GradNorm†	86.2	13.1	12.8	78.7	23.2	23.3	83.1	29.0	29.3
LNL*	82.6	7.6	7.6	74.1	14.2	14.7	77.6	17.7	19.7
UFaTE*	80.9	6.2	6.1	65.9	14.0	13.4	73.4	14.8	14.8
MGDA-Mean†*	81.9	11.8	11.9	74.4	17.8	16.5	75.1	15.7	14.8
MGDA-UFaTE†*	82.8	9.9	9.5	75.3	16.5	16.0	76.7	13.1	13.2
FairMT	**87.7**	**5.5**	**4.5**	**81.9**	**10.1**	**9.9**	83.8	**6.7**	**6.0**

Civil Comments Toxicity Regression

Method	CCC ↑	LGBTQ+			CCC ↑	Race			CCC ↑	Religion		
		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓
MGDA†	73.4	34.9	12.5	27.4	69.2	42.7	18.2	38.9	**72.0**	39.8	15.8	35.3
GradNorm†	71.5	39.5	14.9	26.8	68.4	41.1	18.8	35.6	69.4	39.9	22.3	38.3
LNL*	67.7	28.3	9.7	23.9	65.3	32.9	15.1	27.9	66.1	29.9	9.5	21.6
UFaTE*	65.9	25.5	7.3	21.7	62.6	27.8	15.2	26.1	63.7	25.4	12.8	24.3
MGDA-Mean†*	67.6	31.1	11.2	24.3	63.7	36.7	16.2	31.9	64.4	33.9	12.3	31.3
MGDA-UFaTE†*	68.9	24.9	6.0	22.8	64.1	29.2	14.5	27.3	66.5	28.4	11.8	23.3
FairMT	**73.7**	**21.7**	**4.9**	**19.5**	**69.3**	**23.9**	**10.2**	**21.1**	71.1	**20.7**	**7.2**	**19.9**

† Utility-oriented MTL * Fairness-oriented †* Hybrid fair-MTL.

Table 3: Direct baseline comparison on CelebA (Gender).

Method	Acc ↑	EOD ↓	EO ↓
L2T-FMT	85.7	9.5	7.2
FairGrad-MT	84.9	10.3	7.9
GroupDRO-MT	85.2	11.1	8.4
FairMT	**88.1**	**6.4**	**3.9**

The two components complement each other across output types. Full FairMT achieves higher utility and lower disparities than either single-component variant on all output families in Civil Comments (Race) and partially supervised Affective Analysis (Age) (Tables 15 and 16).

Constraint and calibration sensitivity. The tolerance determines when a batch's aggregate violation prompts an increase in fairness pressure. These updates occur when the batch has an active fairness comparison, which holds for almost every step in the replayed training streams (Appendix K.5). For discrete coordinates, a valid FPR comparison can keep a coordinate active even when its TPR component is unavailable. In matched comparisons, relaxing the tolerance reduces upward multiplier updates across all three benchmarks, but they remain frequent on Civil Comments (Appendix K.2).

Coordinate scales determine the relative size of violations across outputs. On Affective Analysis (Age), using either validation half to estimate the scales gives similar results. The aggregate tolerance is recalibrated from the corresponding validation reference for every scale setting, with other training and model-selection settings held fixed. Moderate perturbations preserve much of the allocation ranking at the calibration checkpoint but increase final EOD across all output families. AU detection EOD rises from 8.8 to 10.9, illustrating that the magnitudes of normalized violations matter alongside their order. Severe distortion and unit scales degrade both utility and fairness in this study (Appendix K.6). The allocation regularizer controls how broadly pressure is shared across active normalized fairness coordinates, while the multiplier sets its overall strength. On CelebA (Gender), intermediate allocation concentration gives the best utility and fairness in the regularization sweep (Appendix H.3).

Trade-offs across tasks. The utility-only sweep in Fig. 2 shifts predictive performance between AU2 detection and Arousal regression while leaving substantial group disparities. Adding AHFDA to the same weight grid moves the evaluated points towards lower AU2 EOD and Arousal KS, giving the update a way to address disparities across both outputs. Both sweeps use the same two-task architecture and data split (Appendix H.1). FairMT attains the highest AU2 F1 and lowest AU2 EOD among the plotted configurations. Relative to AHFDA-MGDA, it also lowers Arousal KS at a small cost in Arousal CCC.

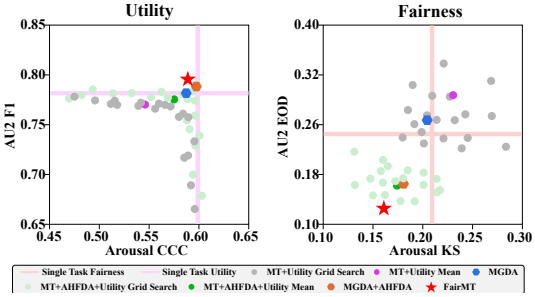

Figure 2: **Pareto analysis on AU2 detection and Arousal regression.** Higher AU2 F1 and Arousal CCC indicate better utility. Lower AU2 EOD and Arousal KS indicate better fairness. The red star denotes full FairMT.

The solver and timing studies assess the cost of coordination. In the CelebA sweep, both inner solves improve from their smallest budgets to the defaults of 20 allocation and 15 task-weight iterations. Increasing either budget further yields no consistent gain across metrics (Appendix H.3). On CelebA (Gender), full FairMT takes about 15% longer per epoch than vanilla MT on one A100, but less time than MGDA and AHFDA-MGDA. Its task-weight update computes alignment statistics from head gradients, avoiding a separate shared-parameter gradient for each utility task. FairMT also has an earlier validation-selected checkpoint than vanilla MT. Both use the same encoder and task heads at inference, so the added solver work is incurred during training (Appendix H.4).

Intersectional grouping. On CelebA, we retain the 29 attribute coordinates and define four Age$\times$Gender groups. Each coordinate compares all four groups to capture gender gaps within each age group and age gaps within each gender (Appendix H.5). FairMT has the highest Accuracy and lowest EOD and EO (Table 4).

Table 4: **CelebA Age$\times$Gender grouping.**

Method	Acc ↑	EOD ↓	EO ↓
LNL	71.2	17.2	14.5
UFaTE	70.9	16.5	14.0
MGDA-Mean	70.1	18.1	16.8
MGDA-UFaTE	72.8	15.2	13.4
FairMT	**77.3**	**9.2**	**6.8**

Ethics and limitations. FairMT addresses algorithmic group fairness through the disparity criteria studied in this work. Improvements in these criteria do not guarantee fair treatment or equitable outcomes in practical use. Real-world fairness also depends on data collection, decision policies, and how predictions affect individuals and communities. Responsible deployment should therefore combine context-specific impact assessment, accountable human oversight, and appropriate institutional and regulatory supervision. Ongoing monitoring should examine actual outcomes across groups, with clear mechanisms to investigate concerns raised by affected individuals, contest decisions, and correct harmful practices. Sensitive group annotations should be handled with appropriate privacy, consent, and data-governance safeguards [38]. FairMT provides algorithmic support for this process; responsibility for fair operation remains with the people and institutions deploying the system.

5 Conclusion

We introduced FairMT, a fairness-constrained framework for heterogeneous multi-task learning. FairMT represents output-specific disparities as fairness coordinates, allocates pressure across them through AHFDA to form a shared constraint, and uses task-head geometry to guide the primal–dual update. Across vision and language benchmarks, FairMT reduces group-level disparities while achieving task utility comparable to that of utility-oriented MTL optimizers.

Acknowledgments and Disclosure of Funding

This work is supported by Queen Mary University of London and the University of Exeter. Guanyu Hu is currently a postdoctoral fellow at the Helsinki Institute for Information Technology (HIIT).

References

[1] Hamid Afshari, Warren Hare, and Solomon Tesfamariam. Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures. *Applied Soft Computing*, 83:105631, 2019.

[2] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International conference on machine learning*, pages 60–69. PMLR, 2018.

[3] Alekh Agarwal, Miroslav Dudík, and Zhiwei Steven Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International conference on machine learning*, pages 120–129. PMLR, 2019.

[4] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

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

[6] Maarten Buyl, MaryBeth Defrance, and Tijl De Bie. fairret: a framework for differentiable fairness regularization terms. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Learning Representations*, volume 2024, pages 22947–22974, 2024. URL `https://proceedings.iclr.cc/paper_files/paper/2024/file/63943ee9fe347f3d95892cf87d9a42e6-Paper-Conference.pdf`.

[7] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[8] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018.

[9] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression via plug-in estimator and recalibration with statistical guarantees. *Advances in Neural Information Processing Systems*, 33:19137–19148, 2020.

[10] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression with wasserstein barycenters. *Advances in Neural Information Processing Systems*, 33:7321–7331, 2020.

[11] John M Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

[12] Sepehr Dehdashtian, Bashir Sadeghi, and Vishnu Naresh Boddeti. Utility-fairness trade-offs and how to find them. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12037–12046, 2024.

[13] Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012.

[14] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[15] C Fabian Benitez-Quiroz, Ramprakash Srinivasan, and Aleix M Martinez. Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5562–5570, 2016.

[16] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[17] Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Kkt conditions, first-order and second-order optimization, and distributed optimization: tutorial and survey. *arXiv preprint arXiv:2110.01858*, 2021.

[18] Yufei Guo, Yinan Wu, Haoran Duan, Guiguang Ding, and Jungong Han. Phaselora: Control-regime-conditioned low-rank adaptation for continuous-action vision-language-action policies. *arXiv preprint arXiv:2608.15285*, 2026.

[19] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

[20] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.

[21] Guanyu Hu, Eleni Papadopoulou, Dimitrios Kollias, Paraskevi Tzouveli, Jie Wei, and Xinyu Yang. Bridging the gap: Protocol towards fair and consistent affect analysis. In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–9, 2024. doi: 10.1109/FG59268.2024.10582033.

[22] Guanyu Hu, Jie Wei, Siyang Song, Dimitrios Kollias, Xinyu Yang, Zhonglin Sun, and Odysseus Kaloidas. Robust facial reactions generation: An emotion-aware framework with modality compensation. In *2024 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10, 2024. doi: 10.1109/IJCB62174.2024.10744499.

[23] Guanyu Hu, Dimitrios Kollias, Eleni Papadopoulou, Paraskevi Tzouveli, Jie Wei, and Xinyu Yang. Rethinking affect analysis: A protocol for ensuring fairness and consistency. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 7(4):914–923, 2025. doi: 10.1109/TBIOM.2025.3550000.

[24] Guanyu Hu, Dimitrios Kollias, and Xinyu Yang. Grounding emotion recognition with visual prototypes: Vega - revisiting clip in merc. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, page 5667–5676, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400720352. doi: 10.1145/3746027.3755340. URL https://doi.org/10.1145/3746027.3755340.

[25] Guanyu Hu, Tangzheng Lian, Dimitrios Kollias, Oya Celiktutan, and Xinyu Yang. Causalaffect: Causal discovery for facial affective understanding. *arXiv preprint arXiv:2512.00456*, 2025.

[26] Guanyu Hu, Dimitrios Kollias, and Xinyu Yang. AURA: Visually interpretable affective understanding via robust archetypes. In Tong Zhang, Miroslav Dudik, Martin Jaggi, Alekh Agarwal, Sharon Li, Dale Schuurmans, Jerry Zhu, Felix Berkenkamp, Hanze Dong, and Alberto Bietti, editors, *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 45181–45211. PMLR, 06–11 Jul 2026. URL https://proceedings.mlr.press/v306/hu26z.html.

[27] Guanyu Hu, Dimitrios Kollias, and Xinyu Yang. From cognitive priors to instance semantics: A unified framework for multi-task affective computing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8551–8562, March 2026.

[28] Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International conference on machine learning*, pages 427–435. PMLR, 2013.

[29] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9012–9020, 2019.

[30] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pages 5637–5664. PMLR, 2021.

[31] Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International conference on machine learning*, pages 2737–2746. PMLR, 2018.

[32] Tangzheng Lian, Guanyu Hu, Dimitrios Kollias, Xinyu Yang, and Oya Celiktutan. Fair domain generalization: An information-theoretic view. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(28):23382–23390, Mar. 2026. doi: 10.1609/aaai.v40i28.39508. URL `https://ojs.aaai.org/index.php/AAAI/article/view/39508`.

[33] Tangzheng Lian, Guanyu Hu, Yijing Ren, Dimitrios Kollias, and Oya Celiktutan. A closed-form solution for debiasing vision-language models with utility guarantees across modalities and tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 31672–31682, June 2026.

[34] Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhang, and Sam Kwong. Pareto multi-task learning. *Advances in neural information processing systems*, 32, 2019.

[35] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[36] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[37] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[38] Xiaoran Lu, Chen Yang, Lu Liang, Guanyu Hu, Ziyi Zhong, and Zihao Jiang. Artificial intelligence for optimizing recruitment and retention in clinical trials: a scoping review. *Journal of the American Medical Informatics Association*, 31(11):2749–2759, 11 2024. ISSN 1527-974X. doi: 10.1093/jamia/ocae243. URL `https://doi.org/10.1093/jamia/ocae243`.

[39] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, pages 3384–3393. PMLR, 2018.

[40] Gaurav Maheshwari and Michaël Perrot. Fairgrad: Fairness aware gradient descent. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=0f8tU3QwWD`.

[41] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, accountability and transparency*, pages 107–118. PMLR, 2018.

[42] Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017.

[43] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan

Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

[44] Taeeon Park, Sangwon Jung, Sanghyuk Chun, and Taesup Moon. Fairdro: Group fairness regularization via classwise robust optimization. *Neural Networks*, 182:106891, 2025. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2024.106891. URL https://www.sciencedirect.com/science/article/pii/S0893608024008207.

[45] Arjun Roy and Eirini Ntoutsi. Learning to teach fairness-aware deep multi-task learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 710–726. Springer, 2022.

[46] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[47] Shiori Sagawa, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

[48] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

[49] Weiran Wang and Miguel A Carreira-Perpinán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.

[50] Yuyan Wang, Xuezhi Wang, Alex Beutel, Flavien Prost, Jilin Chen, and Ed H Chi. Understanding and improving fairness-accuracy trade-offs in multi-task learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1748–1757, 2021.

[51] Hui Wei, Hao Yu, Hui Kuurila-Zhang, and Guoying Zhao. How private is private? a comparative study for face de-identification. *arXiv preprint arXiv:2610.10334*, 2026.

[52] Hui Wei, Hao Yu, and Guoying Zhao. Face de-identification: A domain-centric survey from capture to processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[53] Jie Wei, Guanyu Hu, Xinyu Yang, Anh Tuan Luu, and Yizhuo Dong. Audio-visual domain adaptation feature fusion for speech emotion recognition. In *INTERSPEECH*, pages 1988–1992, 2022.

[54] Jie Wei, Guanyu Hu, Luu Anh Tuan, Xinyu Yang, and Wenjing Zhu. Multi-scale receptive field graph model for emotion recognition in conversations. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10094596.

[55] Jie Wei, Guanyu Hu, Xinyu Yang, Anh Tuan Luu, and Yizhuo Dong. Learning facial expression and body gesture visual information for video emotion recognition. *Expert Systems with Applications*, 237:121419, 2024. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2023.121419. URL https://www.sciencedirect.com/science/article/pii/S0957417423019218.

[56] Shaokui Wei, Jiayin Liu, Bing Li, and Hongyuan Zha. Mean parity fair regression in rkhs. In *International conference on artificial intelligence and statistics*, pages 4602–4628. PMLR, 2023.

[57] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33:5824–5836, 2020.

[58] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pages 1171–1180, 2017.

[59] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P. Gummadi. Fairness Constraints: Mechanisms for Fair Classification. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR, 20–22 Apr 2017. URL `https://proceedings.mlr.press/v54/zafar17a.html`.

[60] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42, 2019.

[61] Yu Zhang and Qiang Yang. An overview of multi-task learning. *National Science Review*, 5(1): 30–43, 2018.

[62] Chen Zhao and Feng Chen. Rank-based multi-task learning for fair regression. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 916–925. IEEE, 2019.

A Active Coordinates and Coordinate Normalization

This appendix specifies the active-coordinate and coordinate-normalization conventions used by FairMT in stochastic training. These conventions determine when a mini-batch provides a valid fairness estimate, how heterogeneous raw violations are normalized before aggregation, and how the AHFDA allocation is converted into gradients on raw fairness coordinates. The detailed construction of discrete and regression asymmetric coordinates is given in Appendix B.

A.1 Parent-task masks and coordinate–group cells

Each fairness coordinate $q \in \mathcal{Q}$ has a parent utility task $t(q) \in \mathcal{T}$. Therefore, the coordinate inherits the observation mask of its parent task:

$$m_{i,q} = m_{i,t(q)}.$$

For a fixed fairness evaluation, the group label is induced by one grouping map,

$$s_i = \pi(a_i) \in \mathcal{G},$$

where $\mathcal{G}$ is the group set for the selected sensitive attribute or intersectional grouping. The grouping map is fixed during a given fairness evaluation and is shared by all utility tasks and fairness coordinates. For a mini-batch $\mathcal{B}$, the observed coordinate–group cell is

$$\mathcal{B}_{g,q} = \{i \in \mathcal{B} : s_i = g,\ m_{i,q} = 1\}. \tag{9}$$

All group-wise fairness statistics are computed only from observed coordinate–group cells. Thus, if a label for the parent task $t(q)$ is missing, the sample contributes neither to the utility loss of $t(q)$ nor to the fairness statistic of coordinate q.

For discrete coordinates, the fairness statistic uses one-vs-rest binary targets. Let $y_{i,q}^{\mathrm{bin}} \in \{0,1\}$ denote the induced binary label for coordinate q, as defined in Appendix B. The positive and negative observed cells are

$$\mathcal{B}_{g,q}^{+} = \left\{i \in \mathcal{B} : s_i = g,\ m_{i,q} = 1,\ y_{i,q}^{\mathrm{bin}} = 1\right\},$$
$$\mathcal{B}_{g,q}^{-} = \left\{i \in \mathcal{B} : s_i = g,\ m_{i,q} = 1,\ y_{i,q}^{\mathrm{bin}} = 0\right\}.$$

The corresponding valid group sets are

$$\mathcal{G}_{q,\mathcal{B}}^{+} = \left\{g \in \mathcal{G} : |\mathcal{B}_{g,q}^{+}| > 0\right\}, \qquad \mathcal{G}_{q,\mathcal{B}}^{-} = \left\{g \in \mathcal{G} : |\mathcal{B}_{g,q}^{-}| > 0\right\}.$$

For regression coordinates, the valid group set is

$$\mathcal{G}_{q,\mathcal{B}}^{\mathrm{reg}} = \{g \in \mathcal{G} : |\mathcal{B}_{g,q}| > 0\}.$$

A.2 Mini-batch active-coordinate convention

A fairness coordinate contributes to the mini-batch fairness signal only when the group comparisons required by its statistic are observable. This active-coordinate convention prevents ill-defined group gaps from entering AHFDA.

For a discrete coordinate $q \in \mathcal{Q}_{\mathrm{bin}} \cup \mathcal{Q}_{\mathrm{cls}}$, the TPR and FPR components are checked separately:

$$I_{q,\mathcal{B}}^{+} = \mathbb{1}\left[|\mathcal{G}_{q,\mathcal{B}}^{+}| \geq 2\right], \qquad I_{q,\mathcal{B}}^{-} = \mathbb{1}\left[|\mathcal{G}_{q,\mathcal{B}}^{-}| \geq 2\right].$$

The active-component denominator for the discrete asymmetric coordinate is

$$Z_{q,\mathcal{B}} = I_{q,\mathcal{B}}^{+} + \beta I_{q,\mathcal{B}}^{-}, \tag{10}$$

where $\beta = 0$ gives an equal-opportunity-style coordinate and $\beta = 1$ gives an equalized-odds-style coordinate. A discrete coordinate is active in mini-batch $\mathcal{B}$ if and only if

$$\mathrm{Valid}_{q,\mathcal{B}}^{\mathrm{disc}} = \mathbb{1}\left[Z_{q,\mathcal{B}} > 0\right] = 1.$$

Thus, when $\beta = 0$, a coordinate is active only if its TPR component is valid; when $\beta = 1$, a coordinate is active if at least one of its TPR or FPR components is valid. The average includes only valid components with positive weight.

For a regression coordinate $q \in \mathcal{Q}_{\mathrm{reg}}$, at least two valid groups are required:

$$\mathrm{Valid}_{q,\mathcal{B}}^{\mathrm{reg}} = \mathbb{1}\left[|\mathcal{G}_{q,\mathcal{B}}^{\mathrm{reg}}| \geq 2\right].$$

Combining the discrete and regression cases, the active coordinate set is

$$\mathcal{Q}_{\mathcal{B}} = \left\{q \in \mathcal{Q}_{\mathrm{bin}} \cup \mathcal{Q}_{\mathrm{cls}} : \mathrm{Valid}_{q,\mathcal{B}}^{\mathrm{disc}} = 1\right\} \cup \left\{q \in \mathcal{Q}_{\mathrm{reg}} : \mathrm{Valid}_{q,\mathcal{B}}^{\mathrm{reg}} = 1\right\}.$$

The number of active coordinates is

$$M_{\mathcal{B}} = |\mathcal{Q}_{\mathcal{B}}|.$$

When $M_{\mathcal{B}} > 0$, the raw active violation vector is

$$\Psi_{\mathcal{B}}(\theta) = (\psi_{q,\mathcal{B}}(\theta))_{q \in \mathcal{Q}_{\mathcal{B}}} \in \mathbb{R}_{\geq 0}^{M_{\mathcal{B}}},$$

where $\psi_{q,\mathcal{B}}$ denotes the mini-batch estimate of the raw asymmetric violation for coordinate q. In the main paper, we suppress the mini-batch subscript and write ψ_q when the active mini-batch context is clear.

If $M_{\mathcal{B}} = 0$, FairMT skips the AHFDA subproblem and shifted aggregate. The dual multiplier is also left unchanged:

$$\lambda^{(r+1)} = \lambda^{(r)} \qquad \text{when} \qquad M_{\mathcal{B}} = 0.$$

In this case, the current update reduces to a utility-only masked MTL update over the active utility tasks in the mini-batch.

A.3 Validation-calibrated coordinate normalization

Raw fairness violations are not directly comparable across output types. Discrete coordinates are squared rate gaps, while regression coordinates depend on the scale of the target and the chosen sample-wise error. To prevent raw numerical scale from determining the global fairness allocation, FairMT applies AHFDA to normalized coordinates.

We first train a utility-only MTL model θ_{MTL} using the same task losses, masks, backbone, and task heads, but without the fairness constraint. Let $\mathcal{V}$ be the set of validation mini-batches. For each coordinate q, let

$$\mathcal{V}_q = \{\mathcal{B} \in \mathcal{V} : q \in \mathcal{Q}_{\mathcal{B}}\}$$

be the set of validation mini-batches in which coordinate q is active. The fixed coordinate scale is

$$\nu_q = \max\left\{\frac{1}{|\mathcal{V}_q|}\sum_{\mathcal{B} \in \mathcal{V}_q} \psi_{q,\mathcal{B}}(\theta_{\mathrm{MTL}}), \nu_{\min}\right\},$$

where $\nu_{\min} > 0$ is a small numerical floor. If $\mathcal{V}_q = \emptyset$, the coordinate can be omitted from constrained training or assigned a type-level fallback scale.

The scales $\{\nu_q\}_{q \in \mathcal{Q}}$ are fixed after calibration and are not updated during FairMT training. For an active mini-batch, the normalized active vector is

$$\bar{\Psi}_{\mathcal{B}}(\theta) = \left(\bar{\psi}_{q,\mathcal{B}}(\theta)\right)_{q \in \mathcal{Q}_{\mathcal{B}}} = \left(\frac{\psi_{q,\mathcal{B}}(\theta)}{\nu_q}\right)_{q \in \mathcal{Q}_{\mathcal{B}}}.$$

AHFDA, the dual residual, and the head-aware fairness sensitivities are computed from $\bar{\Psi}_{\mathcal{B}}$. The allocation therefore reflects the relative sizes of normalized violations.

A.4 Active-set AHFDA input and gradient scaling

When $M_{\mathcal{B}} > 0$, FairMT applies AHFDA to the normalized active vector $\bar{\Psi}_{\mathcal{B}}$. The active-set AHFDA objective is

$$\Phi_{\mu,\mathcal{B}}\left(\bar{\Psi}_{\mathcal{B}}\right) = \max_{w \in \Delta_{M_{\mathcal{B}}}}\left[w^{\top}\bar{\Psi}_{\mathcal{B}} - \frac{\mu}{2}\|w\|_2^2\right],$$

where the active simplex is

$$\Delta_{M_{\mathcal{B}}} = \left\{w \in \mathbb{R}_{\geq 0}^{M_{\mathcal{B}}} : \mathbf{1}^{\top}w = 1\right\}.$$

The shifted active-set aggregate is

$$\widetilde{\Phi}_{\mu,\mathcal{B}}\left(\bar{\Psi}_{\mathcal{B}}\right) = \Phi_{\mu,\mathcal{B}}\left(\bar{\Psi}_{\mathcal{B}}\right) + \frac{\mu}{2M_{\mathcal{B}}}, \tag{11}$$

which satisfies

$$\widetilde{\Phi}_{\mu,\mathcal{B}}(0) = 0 \qquad \text{for the current active set.}$$

The shifted form is used only when $M_{\mathcal{B}} > 0$. When $M_{\mathcal{B}} = 0$, Eq. (11) is not evaluated.

Let $w^{\star}_{\mathcal{B}}$ denote the exact active-set AHFDA allocation:

$$w^{\star}_{\mathcal{B}} = \arg\max_{w \in \Delta_{M_{\mathcal{B}}}} \left[w^{\top}\bar{\Psi}_{\mathcal{B}} - \frac{\mu}{2}\|w\|_2^2 \right].$$

By the water-filling result in Appendix C.1, this allocation is the gradient of the shifted aggregate with respect to the normalized active vector:

$$\nabla_{\bar{\Psi}_{\mathcal{B}}} \widetilde{\Phi}_{\mu,\mathcal{B}}\left(\bar{\Psi}_{\mathcal{B}}\right) = w^{\star}_{\mathcal{B}}. \tag{12}$$

In implementation, FairMT uses the projection-free Frank–Wolfe approximation from Appendix C.2. The final allocation is detached in the outer neural-network update.

The following proposition makes explicit how coordinate normalization affects the gradient received by raw violations.

Proposition 1 (**Gradient scaling under fixed coordinate normalization**)**.** Assume $M_{\mathcal{B}} > 0$ and that the coordinate scales $\{\nu_q\}_{q\in\mathcal{Q}}$ are fixed constants during training. For any active coordinate $q \in \mathcal{Q}_{\mathcal{B}}$, the gradient of the shifted active-set AHFDA aggregate with respect to the raw violation $\psi_{q,\mathcal{B}}$ is

$$\frac{\partial \widetilde{\Phi}_{\mu,\mathcal{B}}\left(\bar{\Psi}_{\mathcal{B}}(\theta)\right)}{\partial \psi_{q,\mathcal{B}}} = \frac{w^{\star}_{q,\mathcal{B}}}{\nu_q}. \tag{13}$$

Proof. For an active coordinate q, the normalized coordinate is

$$\bar{\psi}_{q,\mathcal{B}} = \frac{\psi_{q,\mathcal{B}}}{\nu_q}.$$

Since ν_q is fixed during FairMT training, its derivative is

$$\frac{\partial \bar{\psi}_{q,\mathcal{B}}}{\partial \psi_{q,\mathcal{B}}} = \frac{1}{\nu_q}.$$

From Eq. (12), the derivative of the shifted active-set aggregate with respect to the normalized coordinate is

$$\frac{\partial \widetilde{\Phi}_{\mu,\mathcal{B}}}{\partial \bar{\psi}_{q,\mathcal{B}}} = w^{\star}_{q,\mathcal{B}}.$$

By the chain rule,

$$\frac{\partial \widetilde{\Phi}_{\mu,\mathcal{B}}}{\partial \psi_{q,\mathcal{B}}} = \frac{\partial \widetilde{\Phi}_{\mu,\mathcal{B}}}{\partial \bar{\psi}_{q,\mathcal{B}}} \frac{\partial \bar{\psi}_{q,\mathcal{B}}}{\partial \psi_{q,\mathcal{B}}} = w^{\star}_{q,\mathcal{B}} \frac{1}{\nu_q}.$$

This proves Eq. (13). □

Thus, $w^{\star}_{q,\mathcal{B}}$ determines how much global fairness pressure is assigned to coordinate q in normalized space, while $1/\nu_q$ converts that pressure back to the raw coordinate scale. This convention separates relative violation allocation from raw metric units and ensures that heterogeneous coordinates enter the global constraint through a common normalized interface.

B Detailed Construction of Asymmetric Fairness Coordinates

This appendix gives the full mini-batch construction of the Asymmetric Fairness Coordinates used in Section 3.1. The main text suppresses mini-batch notation for readability; here we write all active sets and estimators with an explicit mini-batch index $\mathcal{B}$. The coordinates defined in this section are raw coordinates ψ_q. They are normalized by the fixed coordinate scales ν_q before entering AHFDA, as described in Appendix A.

B.1 Stop-gradient convention and local surrogate

Asymmetric Fairness Coordinates use detached one-sided references. For any differentiable scalar statistic $u(\theta)$, the stop-gradient operator is defined by

$$\mathrm{sg}(u(\theta)) = u(\theta) \quad \text{in the forward pass,}$$

and

$$\nabla_\theta \,\mathrm{sg}(u(\theta)) = 0 \quad \text{in the backward pass.}$$

Thus, the reference statistic contributes to the numerical value of the violation, but the explicit fairness-gradient path does not differentiate through the group that defines the current reference. FairMT therefore optimizes a local detached-reference surrogate at each update.

For a coordinate q, the parent utility task is denoted by $t(q)$, and the coordinate inherits its observation mask:

$$m_{i,q} = m_{i,t(q)}.$$

All coordinate statistics below are computed only on samples satisfying $m_{i,q} = 1$.

B.2 Discrete coordinates

Discrete coordinates include binary detection tasks and one-vs-rest coordinates induced by multi-class classification tasks. Each discrete coordinate defines a binary target $y^{\mathrm{bin}}_{i,q} \in \{0,1\}$ and a soft predicted-positive score $p_{i,q} \in [0,1]$.

For a binary detection task t, the coordinate is $q = (t)$. Let

$$z_{i,t} = h^{(t)}_{\theta_t}\left(f_{\theta_s}(x_i)\right) \in \mathbb{R}$$

be the task logit. The binary coordinate is defined by

$$y^{\mathrm{bin}}_{i,q} = y_{i,t}, \qquad p_{i,q} = \sigma(z_{i,t}).$$

Here σ is the sigmoid function.

For a multi-class classification task t with C_t classes, each class $c \in \{1,\dots,C_t\}$ induces a one-vs-rest coordinate $q = (t,c)$. Let

$$z_{i,t} = h^{(t)}_{\theta_t}\left(f_{\theta_s}(x_i)\right) \in \mathbb{R}^{C_t}$$

be the class-logit vector. The one-vs-rest coordinate is defined by

$$y^{\mathrm{bin}}_{i,q} = \mathbb{1}[y_{i,t} = c], \qquad p_{i,q} = \mathrm{softmax}(z_{i,t})_c.$$

Therefore, binary detection and multi-class classification share the same coordinate-level discrete fairness estimator after the one-vs-rest transformation.

B.2.1 Soft group-wise rates

For group $g \in \mathcal{G}$, coordinate q, and mini-batch $\mathcal{B}$, define the observed positive and negative coordinate–group cells as

$$\mathcal{B}^+_{g,q} = \left\{ i \in \mathcal{B} : s_i = g,\, m_{i,q} = 1,\, y^{\mathrm{bin}}_{i,q} = 1 \right\},$$

and

$$\mathcal{B}^-_{g,q} = \left\{ i \in \mathcal{B} : s_i = g,\, m_{i,q} = 1,\, y^{\mathrm{bin}}_{i,q} = 0 \right\}.$$

The corresponding valid group sets are

$$\mathcal{G}^+_{q,\mathcal{B}} = \left\{ g \in \mathcal{G} : |\mathcal{B}^+_{g,q}| > 0 \right\},$$

and

$$\mathcal{G}^-_{q,\mathcal{B}} = \left\{ g \in \mathcal{G} : |\mathcal{B}^-_{g,q}| > 0 \right\}.$$

For $g \in \mathcal{G}^+_{q,\mathcal{B}}$, the soft true-positive rate is

$$\widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} = \frac{1}{|\mathcal{B}^+_{g,q}|} \sum_{i \in \mathcal{B}^+_{g,q}} p_{i,q}.$$

For $g \in \mathcal{G}^-_{q,\mathcal{B}}$, the soft false-positive rate is

$$\widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} = \frac{1}{|\mathcal{B}^-_{g,q}|} \sum_{i \in \mathcal{B}^-_{g,q}} p_{i,q}.$$

Using soft scores makes these estimators differentiable with respect to the model outputs. Thresholded predictions are used only for evaluation metrics in Appendix F.2.

B.2.2 Valid components

The TPR and FPR components are checked separately. Define

$$I^+_{q,\mathcal{B}} = \mathbb{1}\left[|\mathcal{G}^+_{q,\mathcal{B}}| \geq 2\right],$$

and

$$I^-_{q,\mathcal{B}} = \mathbb{1}\left[|\mathcal{G}^-_{q,\mathcal{B}}| \geq 2\right].$$

Thus, the TPR component is valid only when at least two groups have observed positive samples for coordinate q, and the FPR component is valid only when at least two groups have observed negative samples.

B.2.3 Detached one-sided references

When $I^+_{q,\mathcal{B}} = 1$, FairMT uses the detached best valid TPR as the TPR reference:

$$\tau_{q,\mathcal{B}} = \mathrm{sg}\left(\max_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q}\right). \tag{14}$$

The max-based one-sided TPR violation is

$$A^{\mathrm{TPR}}_{q,\mathcal{B}} = \max_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \left[\tau_{q,\mathcal{B}} - \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q}\right]^2_+.$$

When $I^+_{q,\mathcal{B}} = 0$, the TPR component is omitted and we set

$$A^{\mathrm{TPR}}_{q,\mathcal{B}} = 0.$$

When $I^-_{q,\mathcal{B}} = 1$, FairMT uses the detached lowest valid FPR as the FPR reference:

$$\phi_{q,\mathcal{B}} = \mathrm{sg}\left(\min_{g \in \mathcal{G}^-_{q,\mathcal{B}}} \widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q}\right).$$

The max-based one-sided FPR violation is

$$A^{\mathrm{FPR}}_{q,\mathcal{B}} = \max_{g \in \mathcal{G}^-_{q,\mathcal{B}}} \left[\widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} - \phi_{q,\mathcal{B}}\right]^2_+.$$

When $I^-_{q,\mathcal{B}} = 0$, the FPR component is omitted and we set

$$A^{\mathrm{FPR}}_{q,\mathcal{B}} = 0.$$

B.2.4 Raw discrete asymmetric violation

Let

$$Z_{q,\mathcal{B}} = I^+_{q,\mathcal{B}} + \beta I^-_{q,\mathcal{B}},$$

where $\beta \geq 0$ controls the relative contribution of the FPR component. If $Z_{q,\mathcal{B}} > 0$, the raw discrete asymmetric violation is

$$\psi^{\mathrm{disc}}_{q,\mathcal{B}}(\theta) = \frac{I^+_{q,\mathcal{B}} A^{\mathrm{TPR}}_{q,\mathcal{B}} + \beta I^-_{q,\mathcal{B}} A^{\mathrm{FPR}}_{q,\mathcal{B}}}{Z_{q,\mathcal{B}}}.$$

If $Z_{q,\mathcal{B}} = 0$, coordinate q is not active in the current mini-batch and is omitted from the active fairness vector:

$$q \in \mathcal{Q}_{\mathcal{B}}^{\text{disc}} \quad \Longleftrightarrow \quad Z_{q,\mathcal{B}} > 0.$$

Here $\mathcal{Q}_{\mathcal{B}}^{\text{disc}}$ is the set of active discrete coordinates. The normalization by $Z_{q,\mathcal{B}}$ prevents a valid component from being artificially shrunk when the other component is unavailable in the current mini-batch. Setting $\beta = 0$ gives an equal-opportunity-style coordinate using only the TPR component. Setting $\beta = 1$ gives an equalized-odds-style coordinate using both TPR and FPR whenever they are observable.

B.3 Regression coordinates

For a scalar regression task t, the corresponding fairness coordinate is $q = (t)$. The model prediction is

$$\hat{y}_{i,q} = h_{\theta_t}^{(t)}\left(f_{\theta_s}(x_i)\right) \in \mathbb{R}.$$

FairMT uses a decomposable sample-wise regression error $\ell_{\text{reg}}(\hat{y}_{i,q}, y_{i,t})$ for the fairness coordinate. In our implementation, ℓ_{reg} is MSE after target normalization. This is separate from the regression utility loss, which may additionally include CCC, as described in Appendix F.4.

For regression, denote the observed coordinate–group cell from Eq. (9) by $\mathcal{B}_{g,q}^{\text{reg}}$:

$$\mathcal{B}_{g,q}^{\text{reg}} = \{i \in \mathcal{B} : s_i = g,\, m_{i,q} = 1\}.$$

The valid regression group set is

$$\mathcal{G}_{q,\mathcal{B}}^{\text{reg}} = \left\{g \in \mathcal{G} : |\mathcal{B}_{g,q}^{\text{reg}}| > 0\right\}.$$

For $g \in \mathcal{G}_{q,\mathcal{B}}^{\text{reg}}$, the empirical group-wise regression error is

$$\widehat{e}_{g,q}^{\mathcal{B}} = \frac{1}{|\mathcal{B}_{g,q}^{\text{reg}}|} \sum_{i \in \mathcal{B}_{g,q}^{\text{reg}}} \ell_{\text{reg}}(\hat{y}_{i,q}, y_{i,t}).$$

The regression coordinate is valid only when at least two groups are observed:

$$\text{Valid}_{q,\mathcal{B}}^{\text{reg}} = \mathbb{1}\left[|\mathcal{G}_{q,\mathcal{B}}^{\text{reg}}| \geq 2\right].$$

When $\text{Valid}_{q,\mathcal{B}}^{\text{reg}} = 1$, FairMT uses the detached lowest-error valid group as the reference:

$$e_{q,\mathcal{B}}^{\text{ref}} = \text{sg}\left(\min_{g \in \mathcal{G}_{q,\mathcal{B}}^{\text{reg}}} \widehat{e}_{g,q}^{\mathcal{B}}\right).$$

The raw regression asymmetric violation is

$$\psi_{q,\mathcal{B}}^{\text{reg}}(\theta) = \max_{g \in \mathcal{G}_{q,\mathcal{B}}^{\text{reg}}} \left[\widehat{e}_{g,q}^{\mathcal{B}} - e_{q,\mathcal{B}}^{\text{ref}}\right]_+^2.$$

If $\text{Valid}_{q,\mathcal{B}}^{\text{reg}} = 0$, coordinate q is omitted from the active fairness vector:

$$q \in \mathcal{Q}_{\mathcal{B}}^{\text{reg}} \quad \Longleftrightarrow \quad \text{Valid}_{q,\mathcal{B}}^{\text{reg}} = 1.$$

Here $\mathcal{Q}_{\mathcal{B}}^{\text{reg}}$ is the set of active regression coordinates.

B.4 Forward gap equivalence and asymmetric gradient path

The asymmetric coordinates preserve the forward value of worst group-disparity gaps while changing the backward path through the detached reference. The following proposition makes this explicit.

Proposition 2 (Forward gap equivalence and asymmetric gradient path)**.** Consider an active coordinate q in mini-batch $\mathcal{B}$.

(i) If the TPR component is valid, then the forward value of $A_{q,\mathcal{B}}^{\text{TPR}}$ equals the squared worst pairwise TPR gap:

$$A_{q,\mathcal{B}}^{\text{TPR}} = \left(\max_{g \in \mathcal{G}_{q,\mathcal{B}}^{+}} \widehat{\text{TPR}}_{g,q}^{\mathcal{B}} - \min_{g \in \mathcal{G}_{q,\mathcal{B}}^{+}} \widehat{\text{TPR}}_{g,q}^{\mathcal{B}}\right)^2.$$

(ii) If the FPR component is valid, then the forward value of $A^{\mathrm{FPR}}_{q,\mathcal{B}}$ equals the squared worst pairwise FPR gap:

$$A^{\mathrm{FPR}}_{q,\mathcal{B}} = \left(\max_{g \in \mathcal{G}^-_{q,\mathcal{B}}} \widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} - \min_{g \in \mathcal{G}^-_{q,\mathcal{B}}} \widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} \right)^2 .$$

(iii) If the regression coordinate is valid, then the forward value of $\psi^{\mathrm{reg}}_{q,\mathcal{B}}$ equals the squared worst pairwise group-error gap:

$$\psi^{\mathrm{reg}}_{q,\mathcal{B}} = \left(\max_{g \in \mathcal{G}^{\mathrm{reg}}_{q,\mathcal{B}}} \widehat{e}^{\mathcal{B}}_{g,q} - \min_{g \in \mathcal{G}^{\mathrm{reg}}_{q,\mathcal{B}}} \widehat{e}^{\mathcal{B}}_{g,q} \right)^2 .$$

In the backward pass, however, the explicit subgradient does not differentiate through the reference statistic because the reference is detached.

Proof. For the TPR component, Eq. (14) gives

$$\tau_{q,\mathcal{B}} = \max_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} \quad \text{in the forward pass.}$$

Therefore,

$$\max_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \left[\tau_{q,\mathcal{B}} - \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} \right]^2_+ = \left(\max_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} - \min_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} \right)^2 .$$

The FPR and regression cases follow by the same argument, using the detached minimum FPR reference and detached minimum regression-error reference. □

We now state the corresponding gradient-path interpretation. Define the active maximizer sets

$$\mathcal{W}^{\mathrm{TPR}}_{q,\mathcal{B}} = \operatorname*{argmax}_{g \in \mathcal{G}^+_{q,\mathcal{B}}} \left[\tau_{q,\mathcal{B}} - \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} \right]^2_+ ,$$

$$\mathcal{W}^{\mathrm{FPR}}_{q,\mathcal{B}} = \operatorname*{argmax}_{g \in \mathcal{G}^-_{q,\mathcal{B}}} \left[\widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} - \phi_{q,\mathcal{B}} \right]^2_+ ,$$

and

$$\mathcal{W}^{\mathrm{reg}}_{q,\mathcal{B}} = \operatorname*{argmax}_{g \in \mathcal{G}^{\mathrm{reg}}_{q,\mathcal{B}}} \left[\widehat{e}^{\mathcal{B}}_{g,q} - e^{\mathrm{ref}}_{q,\mathcal{B}} \right]^2_+ .$$

At non-smooth points of the maximum and hinge operations, gradients are understood as subgradients. For the TPR component, the subgradient satisfies

$$\partial_\theta A^{\mathrm{TPR}}_{q,\mathcal{B}} \subseteq \mathrm{conv} \left\{ -2 \left[\tau_{q,\mathcal{B}} - \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} \right]_+ \nabla_\theta \widehat{\mathrm{TPR}}^{\mathcal{B}}_{g,q} : g \in \mathcal{W}^{\mathrm{TPR}}_{q,\mathcal{B}} \right\} . \tag{15}$$

For the FPR component,

$$\partial_\theta A^{\mathrm{FPR}}_{q,\mathcal{B}} \subseteq \mathrm{conv} \left\{ 2 \left[\widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} - \phi_{q,\mathcal{B}} \right]_+ \nabla_\theta \widehat{\mathrm{FPR}}^{\mathcal{B}}_{g,q} : g \in \mathcal{W}^{\mathrm{FPR}}_{q,\mathcal{B}} \right\} .$$

For the regression coordinate,

$$\partial_\theta \psi^{\mathrm{reg}}_{q,\mathcal{B}} \subseteq \mathrm{conv} \left\{ 2 \left[\widehat{e}^{\mathcal{B}}_{g,q} - e^{\mathrm{ref}}_{q,\mathcal{B}} \right]_+ \nabla_\theta \widehat{e}^{\mathcal{B}}_{g,q} : g \in \mathcal{W}^{\mathrm{reg}}_{q,\mathcal{B}} \right\} . \tag{16}$$

Equations (15)–(16) show that explicit fairness-gradient pressure is placed on the currently worst violating valid group or groups for that coordinate. The group defining the detached reference does not receive gradient through the reference value itself. Joint updates to the shared encoder and task heads can indirectly affect all groups, including the reference group.

B.5 Connection to normalized AHFDA input

The constructions above define raw coordinate violations:

$$\psi_{q,\mathcal{B}}(\theta) \in \left\{\psi^{\mathrm{disc}}_{q,\mathcal{B}}(\theta), \psi^{\mathrm{reg}}_{q,\mathcal{B}}(\theta)\right\}.$$

The active raw violation vector in mini-batch $\mathcal{B}$ is

$$\Psi_{\mathcal{B}}(\theta) = \left(\psi_{q,\mathcal{B}}(\theta)\right)_{q\in\mathcal{Q}_{\mathcal{B}}}.$$

FairMT normalizes each active raw coordinate using its fixed validation-calibrated coordinate scale:

$$\bar{\psi}_{q,\mathcal{B}}(\theta) = \frac{\psi_{q,\mathcal{B}}(\theta)}{\nu_q}, \qquad q \in \mathcal{Q}_{\mathcal{B}}.$$

The normalized active vector is

$$\bar{\Psi}_{\mathcal{B}}(\theta) = \left(\bar{\psi}_{q,\mathcal{B}}(\theta)\right)_{q\in\mathcal{Q}_{\mathcal{B}}}.$$

AHFDA is applied to $\bar{\Psi}_{\mathcal{B}}(\theta)$, not to $\Psi_{\mathcal{B}}(\theta)$. Therefore, if $\bar{w}_{q,\mathcal{B}}$ is the detached allocation weight for coordinate q, the gradient of the shifted active-set Frank–Wolfe (FW) estimate $\widehat{\Phi}^{\mathrm{FW}}_{\mu,\mathcal{B}}$ with respect to the raw coordinate is

$$\frac{\partial\widehat{\Phi}^{\mathrm{FW}}_{\mu,\mathcal{B}}}{\partial\psi_{q,\mathcal{B}}} = \frac{\bar{w}_{q,\mathcal{B}}}{\nu_q}.$$

Thus, AHFDA allocates pressure according to relative normalized violation size, while the raw coordinate gradient is rescaled by the fixed calibration factor $1/\nu_q$.

C AHFDA Derivations

C.1 Closed-Form Water-Filling Solution of AHFDA

This appendix proves the closed-form allocation used by AHFDA. We state the derivation for a generic normalized active violation vector. In the full-coordinate case, the dimension is $d = M$ and the vector is $v = \bar{\Psi}(\theta)$. In a stochastic mini-batch, if the active coordinate set is

$$\mathcal{Q}_{\mathcal{B}} = \{q_1, \ldots, q_d\}, \qquad d = M_{\mathcal{B}},$$

then the generic vector is

$$v = (v_1, \ldots, v_d) = (\bar{\psi}_{q_1}, \ldots, \bar{\psi}_{q_d}) = \left(\frac{\psi_{q_1}}{\nu_{q_1}}, \ldots, \frac{\psi_{q_d}}{\nu_{q_d}}\right).$$

Throughout this section we assume $d \geq 1$. If $M_{\mathcal{B}} = 0$, no active-set AHFDA subproblem is formed, no fairness-gradient signal is produced, and the dual multiplier is left unchanged.

For $v \in \mathbb{R}^d_{\geq 0}$ and $\mu > 0$, define the active-set AHFDA objective

$$\Phi_{\mu,d}(v) = \max_{w\in\Delta_d}\left[w^\top v - \frac{\mu}{2}\|w\|_2^2\right], \tag{17}$$

where

$$\Delta_d = \left\{w \in \mathbb{R}^d_{\geq 0} : \mathbf{1}^\top w = 1\right\}.$$

The shifted aggregate is

$$\widetilde{\Phi}_{\mu,d}(v) = \Phi_{\mu,d}(v) + \frac{\mu}{2d}. \tag{18}$$

C.1.1 Zero baseline and shifted aggregate

The shift in Eq. (18) makes the aggregate vanish when all active normalized violations are zero. To see this, set $v = 0$. Then Eq. (17) becomes

$$\Phi_{\mu,d}(0) = \max_{w\in\Delta_d}\left[-\frac{\mu}{2}\|w\|_2^2\right] = -\frac{\mu}{2}\min_{w\in\Delta_d}\|w\|_2^2.$$

The minimum-norm point on the simplex is the uniform vector

$$u_d = \frac{1}{d}\mathbf{1},$$

whose squared norm is

$$\|u_d\|_2^2 = \frac{1}{d}.$$

Therefore,

$$\Phi_{\mu,d}(0) = -\frac{\mu}{2d}, \qquad \widetilde{\Phi}_{\mu,d}(0) = 0.$$

Thus, the shifted aggregate has the natural interpretation of a nonnegative global violation level relative to the zero-violation baseline.

C.1.2 Projection equivalence and uniqueness

For fixed v, the objective in Eq. (17) is μ-strongly concave in w, and the simplex Δ_d is compact and convex. Therefore, the maximizer exists and is unique. The maximization is equivalent to the convex minimization

$$\min_{w\in\Delta_d}\left[\frac{\mu}{2}\|w\|_2^2 - v^\top w\right].$$

Completing the square gives

$$\frac{\mu}{2}\|w\|_2^2 - v^\top w = \frac{\mu}{2}\left\|w - \frac{v}{\mu}\right\|_2^2 - \frac{1}{2\mu}\|v\|_2^2. \tag{19}$$

The second term in Eq. (19) is independent of w. Hence the unique maximizer of Eq. (17) is the Euclidean projection of v/μ onto the simplex:

$$w^\star(v) = \Pi_{\Delta_d}\left(\frac{v}{\mu}\right). \tag{20}$$

C.1.3 Water-filling Karush–Kuhn–Tucker (KKT) characterization

We now derive the coordinate-wise water-filling form of Eq. (20). Let

$$z = \frac{v}{\mu}.$$

The projection problem is

$$\min_{w\in\mathbb{R}^d}\frac{1}{2}\|w - z\|_2^2 \quad \text{s.t.} \quad \mathbf{1}^\top w = 1, \quad w \geq 0.$$

Introduce a scalar multiplier $\kappa \in \mathbb{R}$ for the equality constraint and nonnegative multipliers $\xi_j \geq 0$ for the inequality constraints. The Lagrangian is

$$\mathcal{L}(w,\kappa,\xi) = \frac{1}{2}\sum_{j=1}^{d}(w_j - z_j)^2 + \kappa\left(\sum_{j=1}^{d} w_j - 1\right) - \sum_{j=1}^{d}\xi_j w_j.$$

The KKT stationarity condition [17] is

$$\frac{\partial\mathcal{L}}{\partial w_j} = w_j - z_j + \kappa - \xi_j = 0, \qquad j = 1,\ldots,d. \tag{21}$$

Primal feasibility, dual feasibility, and complementary slackness require

$$\sum_{j=1}^{d} w_j = 1, \qquad w_j \geq 0, \qquad \xi_j \geq 0, \qquad \xi_j w_j = 0.$$

If $w_j^\star > 0$, then complementary slackness gives $\xi_j = 0$, and Eq. (21) gives

$$w_j^\star = z_j - \kappa.$$

If $w_j^\star = 0$, then Eq. (21) gives

$$\xi_j = \kappa - z_j,$$

and dual feasibility implies

$$z_j - \kappa \le 0.$$

Combining the two cases gives the water-filling form

$$w_j^\star(v) = [z_j - \kappa]_+ = \left[\frac{v_j}{\mu} - \kappa\right]_+, \qquad j = 1, \dots, d,$$

where κ enforces the simplex constraint:

$$\sum_{j=1}^{d} \left[\frac{v_j}{\mu} - \kappa\right]_+ = 1. \tag{22}$$

The left-hand side of Eq. (22) is continuous and strictly decreasing where at least one summand is positive, with value tending to $+\infty$ as $\kappa \to -\infty$ and value 0 when $\kappa \ge \max_j v_j/\mu$. Therefore, the threshold κ is uniquely determined. Coordinates below the threshold receive zero allocation, while coordinates above the threshold receive allocation proportional to their excess over the threshold.

C.1.4 Differentiability and AHFDA gradient

The mapping $v \mapsto \Phi_{\mu,d}(v)$ is differentiable. The inner maximization in Eq. (17) has a compact feasible set and a unique maximizer $w^\star(v)$. Moreover, the maximized objective is differentiable in v, with partial derivative w. By Danskin's theorem,

$$\nabla_v \Phi_{\mu,d}(v) = w^\star(v).$$

The shift in Eq. (18) is constant with respect to v. Therefore,

$$\nabla_v \widetilde{\Phi}_{\mu,d}(v) = w^\star(v). \tag{23}$$

Thus, the AHFDA allocation solves the inner fairness-pressure allocation problem and gives the gradient of the shifted global fairness aggregate with respect to the normalized active violation vector.

C.1.5 Connection to raw-coordinate gradients

AHFDA is applied to normalized coordinates. In a mini-batch with active coordinate set

$$\mathcal{Q}_\mathcal{B} = \{q_1, \dots, q_d\},$$

the active normalized vector satisfies

$$v_j = \bar{\psi}_{q_j} = \frac{\psi_{q_j}}{\nu_{q_j}}, \qquad j = 1, \dots, d. \tag{24}$$

Combining Eq. (23) with Eq. (24), the chain rule gives

$$\frac{\partial \widetilde{\Phi}_{\mu,d}(\bar{\Psi}_\mathcal{B})}{\partial \psi_{q_j}} = \frac{w_j^\star(\bar{\Psi}_\mathcal{B})}{\nu_{q_j}}, \qquad j = 1, \dots, d.$$

Thus, an active coordinate receives fairness-gradient pressure according to two factors: its AHFDA allocation $w_j^\star$, and the inverse of its fixed calibration scale ν_{q_j}. Inactive coordinates are not part of the active-set vector and receive no AHFDA gradient in that mini-batch.

C.1.6 Mean–max interpolation property

The shifted AHFDA aggregate lies between the uniform mean violation and the hard maximum violation. This formalizes its role as a regularized worst-coordinate aggregator.

Proposition 3 (Mean–max bounds)**.** For any $v \in \mathbb{R}^d_{\ge 0}$ and $\mu > 0$,

$$\frac{1}{d}\mathbf{1}^\top v \le \widetilde{\Phi}_{\mu,d}(v) \le \max_{1 \le j \le d} v_j.$$

Proof. For the lower bound, choose the uniform allocation $u_d = \frac{1}{d}\mathbf{1}$. Since $u_d \in \Delta_d$, the optimal value is at least the value at u_d:

$$\widetilde{\Phi}_{\mu,d}(v) \geq u_d^\top v - \frac{\mu}{2}\|u_d\|_2^2 + \frac{\mu}{2d}.$$

Using $\|u_d\|_2^2 = 1/d$, we obtain

$$\widetilde{\Phi}_{\mu,d}(v) \geq \frac{1}{d}\mathbf{1}^\top v.$$

For the upper bound, let $v_{\max} = \max_j v_j$. For any $w \in \Delta_d$,

$$w^\top v \leq v_{\max}.$$

Also, by Cauchy's inequality,

$$\|w\|_2^2 \geq \frac{1}{d}(\mathbf{1}^\top w)^2 = \frac{1}{d}.$$

Therefore, for every $w \in \Delta_d$,

$$w^\top v - \frac{\mu}{2}\|w\|_2^2 + \frac{\mu}{2d} \leq v_{\max}.$$

Taking the maximum over $w \in \Delta_d$ gives the upper bound. □

The lower bound shows that AHFDA never falls below uniform averaging of active normalized violations. The upper bound shows that it never exceeds the hard worst-coordinate violation. Therefore, AHFDA preserves sensitivity to large violations while avoiding the discontinuous allocation behavior of an unregularized maximum.

C.1.7 Smoothness and dependence on μ

The quadratic regularization also makes the aggregate smooth with respect to normalized violations. From Eq. (20) and Eq. (23),

$$\nabla_v \widetilde{\Phi}_{\mu,d}(v) = \Pi_{\Delta_d}\left(\frac{v}{\mu}\right).$$

Euclidean projection onto a closed convex set is non-expansive. For any $v, v' \in \mathbb{R}^d_{\geq 0}$,

$$\left\|\nabla_v \widetilde{\Phi}_{\mu,d}(v) - \nabla_v \widetilde{\Phi}_{\mu,d}(v')\right\|_2 = \left\|\Pi_{\Delta_d}\left(\frac{v}{\mu}\right) - \Pi_{\Delta_d}\left(\frac{v'}{\mu}\right)\right\|_2 \leq \frac{1}{\mu}\|v - v'\|_2.$$

Thus, $\widetilde{\Phi}_{\mu,d}$ is $1/\mu$-smooth with respect to the normalized active violation vector. Smaller μ makes the allocation sharper and closer to a hard maximum, while larger μ makes the allocation smoother and more spread across active coordinates.

This limiting behavior can be made explicit. Let

$$v_{\max} = \max_{1 \leq j \leq d} v_j.$$

Choosing the unit vector $e_{j^\star}$ with $j^\star \in \arg\max_j v_j$ gives

$$\widetilde{\Phi}_{\mu,d}(v) \geq v_{\max} - \frac{\mu}{2} + \frac{\mu}{2d}.$$

Together with the upper bound in Proposition 3, this gives

$$v_{\max} - \frac{\mu}{2}\left(1 - \frac{1}{d}\right) \leq \widetilde{\Phi}_{\mu,d}(v) \leq v_{\max}.$$

Therefore,

$$\lim_{\mu \to 0^+} \widetilde{\Phi}_{\mu,d}(v) = \max_{1 \leq j \leq d} v_j.$$

When μ is larger and all coordinates remain active in the simplex projection, the water-filling form gives

$$w_j^\star(v) = \frac{1}{d} + \frac{v_j - \bar{v}}{\mu}, \qquad \bar{v} = \frac{1}{d}\mathbf{1}^\top v,$$

provided that the right-hand side is positive for every j. This expression shows how larger μ moves the allocation toward the uniform vector, while still increasing pressure on coordinates whose normalized violations exceed the active-set mean.

C.1.8 Interpretation for FairMT

The closed-form solution shows that AHFDA performs adaptive allocation over normalized fairness coordinates. Group-level directionality comes from the asymmetric coordinate construction in Appendix B. AHFDA decides which active output-side coordinates receive more global constraint pressure. The gradient passed to a raw coordinate q is scaled as $w_q^\star/\nu_q$, with allocation determined by the relative size of the normalized violation.

In summary, AHFDA provides three properties needed by FairMT: it is a smooth surrogate of a worst-coordinate fairness constraint, it interpolates between mean aggregation and hard maximum behavior, and it yields an explicit gradient allocation over active normalized fairness coordinates.

C.2 Projection-Free Optimization for AHFDA

This appendix gives the projection-free solver used to compute the AHFDA allocation during training. The closed-form solution in Appendix C.1 shows that the exact allocation can be obtained by projecting the scaled normalized violation vector onto the simplex. During stochastic neural-network training, a feasible approximate allocation identifies high-violation active coordinates and supplies the outer-gradient weights. We therefore use a Frank–Wolfe [16, 28] solver on the active normalized violation vector. The solver avoids Euclidean simplex projection, preserves feasibility at every inner step, admits a computable optimality certificate, and naturally yields sparse allocations over high-violation coordinates.

C.2.1 Active normalized input and convex reformulation

Consider a mini-batch $\mathcal{B}$ with active fairness-coordinate set

$$\mathcal{Q}_\mathcal{B} = \{q_1, \dots, q_d\}, \qquad d = M_\mathcal{B} = |\mathcal{Q}_\mathcal{B}|.$$

The AHFDA solver is called only when $d > 0$. If $d = 0$, the mini-batch contains no valid fairness comparison, no AHFDA subproblem is formed, and the fairness-gradient and dual updates are skipped. If $d = 1$, the simplex contains a single feasible point, and the allocation is $w = (1)$.

For $d \geq 2$, the solver receives the active normalized violation vector

$$v = \bar{\Psi}_\mathcal{B} = \left(\bar{\psi}_{q_1}, \dots, \bar{\psi}_{q_d}\right) = \left(\frac{\psi_{q_1}}{\nu_{q_1}}, \dots, \frac{\psi_{q_d}}{\nu_{q_d}}\right) \in \mathbb{R}^d_{\geq 0}.$$

Thus, the solver operates on normalized active coordinates, not on raw violations.

For a generic $v \in \mathbb{R}^d_{\geq 0}$, the active-set AHFDA aggregate is

$$\Phi_{\mu,d}(v) = \max_{w \in \Delta_d} \left[w^\top v - \frac{\mu}{2}\|w\|_2^2 \right], \qquad \Delta_d = \{w \in \mathbb{R}^d_{\geq 0} : \mathbf{1}^\top w = 1\}. \tag{25}$$

The shifted aggregate is

$$\widetilde{\Phi}_{\mu,d}(v) = \Phi_{\mu,d}(v) + \frac{\mu}{2d}.$$

The maximization in Eq. (25) is equivalent to the convex quadratic minimization

$$\min_{w \in \Delta_d} f_\mu(w; v), \qquad f_\mu(w; v) = \frac{\mu}{2}\|w\|_2^2 - v^\top w. \tag{26}$$

Let $w^\star(v)$ denote the unique minimizer of Eq. (26). Then

$$\widetilde{\Phi}_{\mu,d}(v) = -f_\mu(w^\star(v); v) + \frac{\mu}{2d}. \tag{27}$$

The gradient of the convex objective with respect to w is

$$\nabla_w f_\mu(w; v) = \mu w - v. \tag{28}$$

Since $f_\mu(\cdot; v)$ is μ-strongly convex and μ-smooth, it has a unique optimizer over the compact simplex Δ_d.

C.2.2 Frank–Wolfe update

Frank–Wolfe minimizes Eq. (26) using only a linear minimization oracle over the simplex. At inner iteration ℓ, given a feasible allocation $w^{(\ell)} \in \Delta_d$, the method solves

$$s^{(\ell)} \in \arg\min_{s \in \Delta_d} \left\langle \nabla_w f_\mu(w^{(\ell)}; v), s \right\rangle.$$

The feasible set is the probability simplex, and the minimum of the linearized objective is attained at a vertex. Using Eq. (28), the selected vertex is

$$s^{(\ell)} = e_{j_\ell}, \qquad j_\ell \in \arg\min_{j \in \{1,\dots,d\}} \left(\mu w_j^{(\ell)} - v_j \right).$$

Equivalently,

$$j_\ell \in \arg\max_{j \in \{1,\dots,d\}} \left(v_j - \mu w_j^{(\ell)} \right). \tag{29}$$

Thus, each Frank–Wolfe step selects the coordinate with the largest adjusted normalized violation. The adjustment term $\mu w_j^{(\ell)}$ discourages repeatedly assigning all mass to a coordinate that already has large allocation.

The Frank–Wolfe direction is

$$p^{(\ell)} = s^{(\ell)} - w^{(\ell)}.$$

With step size γ_ℓ, the allocation is updated by

$$w^{(\ell+1)} = w^{(\ell)} + \gamma_\ell p^{(\ell)} = (1 - \gamma_\ell) w^{(\ell)} + \gamma_\ell s^{(\ell)}, \qquad \gamma_\ell \in [0, 1]. \tag{30}$$

Since both $w^{(\ell)}$ and $s^{(\ell)}$ lie in Δ_d, and $\gamma_\ell \in [0, 1]$, Eq. (30) preserves simplex feasibility:

$$w^{(\ell+1)} \in \Delta_d.$$

Therefore, no Euclidean projection onto the simplex is required.

C.2.3 Exact line search

FairMT uses exact line search along the Frank–Wolfe direction. Define

$$\varphi_\ell(\gamma) = f_\mu \left(w^{(\ell)} + \gamma p^{(\ell)}; v \right), \qquad \gamma \in [0, 1].$$

Using the quadratic form of f_μ, the derivative is

$$\varphi'_\ell(\gamma) = \left\langle \nabla_w f_\mu(w^{(\ell)}; v) + \mu \gamma p^{(\ell)}, p^{(\ell)} \right\rangle.$$

If $p^{(\ell)} \neq 0$, the unconstrained minimizer satisfies $\varphi'_\ell(\gamma) = 0$, which gives

$$\gamma_\ell^{\mathrm{unc}} = -\frac{\left\langle \nabla_w f_\mu(w^{(\ell)}; v), p^{(\ell)} \right\rangle}{\mu \|p^{(\ell)}\|_2^2}.$$

The feasible line-search step is obtained by clipping to $[0, 1]$:

$$\gamma_\ell = \Pi_{[0,1]} \left(-\frac{\left\langle \nabla_w f_\mu(w^{(\ell)}; v), p^{(\ell)} \right\rangle}{\mu \|p^{(\ell)}\|_2^2} \right),$$

where $\Pi_{[0,1]}(a) = \min\{\max\{a, 0\}, 1\}$. If $p^{(\ell)} = 0$, then the current iterate is already a vertex selected by the oracle, and the inner loop stops.

C.2.4 Optimality certificate and convergence

The Frank–Wolfe gap at iterate $w^{(\ell)}$ is

$$\mathcal{G}_{\mathrm{FW}}^{(\ell)} = \left\langle \nabla_w f_\mu(w^{(\ell)}; v), w^{(\ell)} - s^{(\ell)} \right\rangle.$$

This quantity is nonnegative and computable from the same linear minimization oracle used by the solver. It also upper-bounds the primal suboptimality.

Proposition 4 (Frank–Wolfe optimality certificate)**.** Let $w^\star(v)$ be the minimizer of Eq. (26). For every Frank–Wolfe iterate $w^{(\ell)} \in \Delta_d$,

$$0 \le f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v) \le \mathcal{G}_{\mathrm{FW}}^{(\ell)}. \tag{31}$$

Proof. By convexity of $f_\mu(\cdot; v)$,

$$f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v) \le \left\langle \nabla_w f_\mu(w^{(\ell)}; v), w^{(\ell)} - w^\star(v) \right\rangle. \tag{32}$$

Since $s^{(\ell)}$ minimizes the linearized objective over Δ_d,

$$\left\langle \nabla_w f_\mu(w^{(\ell)}; v), s^{(\ell)} \right\rangle \le \left\langle \nabla_w f_\mu(w^{(\ell)}; v), w^\star(v) \right\rangle. \tag{33}$$

Combining Eq. (32) and Eq. (33) yields

$$f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v) \le \left\langle \nabla_w f_\mu(w^{(\ell)}; v), w^{(\ell)} - s^{(\ell)} \right\rangle = \mathcal{G}_{\mathrm{FW}}^{(\ell)}.$$

The nonnegativity follows from the optimality of $w^\star(v)$. □

The standard Frank–Wolfe rate also gives a sublinear convergence guarantee. Since f_μ is μ-smooth and the Euclidean diameter of the simplex is $\sqrt{2}$, the curvature constant satisfies

$$C_f \le 2\mu.$$

With exact line search, the primal gap after $\ell \ge 1$ updates satisfies

$$f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v) \le \frac{2C_f}{\ell + 2} \le \frac{4\mu}{\ell + 2}. \tag{34}$$

The inner solver stops at its iteration budget or when the gap $\mathcal{G}_{\mathrm{FW}}^{(\ell)}$ is at most the tolerance δ_{FW}. In the latter case, Eq. (31) certifies that the allocation is δ_{FW}-optimal for the active AHFDA subproblem.

C.2.5 Approximation error for the AHFDA aggregate

For any feasible allocation $w \in \Delta_d$, define the finite-step AHFDA estimate

$$\widehat{\Phi}_{\mu,d}^{\mathrm{FW}}(v; w) = w^\top v - \frac{\mu}{2}\|w\|_2^2 + \frac{\mu}{2d}.$$

Using Eq. (26), this can be written as

$$\widehat{\Phi}_{\mu,d}^{\mathrm{FW}}(v; w) = -f_\mu(w; v) + \frac{\mu}{2d}. \tag{35}$$

Combining Eq. (27) and Eq. (35), for any Frank–Wolfe iterate $w^{(\ell)}$, we have

$$\widetilde{\Phi}_{\mu,d}(v) - \widehat{\Phi}_{\mu,d}^{\mathrm{FW}}(v; w^{(\ell)}) = f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v).$$

Therefore, the finite-step estimate is a lower approximation of the exact shifted AHFDA aggregate, and its underestimation error is controlled by the Frank–Wolfe primal gap:

$$0 \le \widetilde{\Phi}_{\mu,d}(v) - \widehat{\Phi}_{\mu,d}^{\mathrm{FW}}(v; w^{(\ell)}) \le \mathcal{G}_{\mathrm{FW}}^{(\ell)}.$$

For $\ell \ge 1$, Eq. (34) bounds this error by

$$\widetilde{\Phi}_{\mu,d}(v) - \widehat{\Phi}_{\mu,d}^{\mathrm{FW}}(v; w^{(\ell)}) \le \frac{4\mu}{\ell + 2}.$$

Thus, finite Frank–Wolfe iterations give a controlled projection-free approximation to the exact AHFDA constraint value.

C.2.6 Allocation error and outer-gradient approximation

The μ-strong convexity of $f_\mu(\cdot; v)$ also controls the allocation error through the primal gap. For any iterate $w^{(\ell)}$,

$$\frac{\mu}{2}\left\|w^{(\ell)} - w^\star(v)\right\|_2^2 \le f_\mu(w^{(\ell)}; v) - f_\mu(w^\star(v); v). \tag{36}$$

Combining Eq. (36) with Eq. (31) gives

$$\left\|w^{(\ell)} - w^\star(v)\right\|_2 \le \sqrt{\frac{2\mathcal{G}_{\mathrm{FW}}^{(\ell)}}{\mu}}. \tag{37}$$

Therefore, the same computable Frank–Wolfe gap that certifies objective accuracy also certifies closeness of the finite-step allocation to the exact water-filling allocation. In the exact case, Danskin's theorem [11] gives

$$\nabla_v \widetilde{\Phi}_{\mu,d}(v) = w^\star(v).$$

With finite iterations, FairMT uses the feasible approximate allocation $w^{(\ell)}$ as a projection-free approximation to this gradient. The approximation quality is controlled by Eq. (37).

C.2.7 Sparsity and active-coordinate interpretation

Frank–Wolfe iterates remain sparse when initialized at a simplex vertex. In FairMT, we initialize with the largest normalized violation:

$$j_{\mathrm{init}} \in \arg\max_{j\in\{1,\dots,d\}} v_j, \qquad w^{(0)} = e_{j_{\mathrm{init}}}.$$

Each update is a convex combination of the current iterate and one selected simplex vertex. By induction,

$$w^{(L)} \in \mathrm{conv}\left(\{e_{j_{\mathrm{init}}}\} \cup \{e_{j_0}, e_{j_1}, \dots, e_{j_{L-1}}\}\right).$$

Consequently, the support size is bounded by

$$\|w^{(L)}\|_0 \le L + 1.$$

The support can be smaller if coordinates are selected repeatedly. The support of $w^{(L)}$ is drawn from the initial vertex and the coordinates selected by Eq. (29), which prioritizes large adjusted normalized violations.

C.2.8 Projection-free AHFDA solver

Algorithm 2 summarizes the solver used in FairMT training. The input vector is the active normalized violation vector $v = \bar{\Psi}_{\mathcal{B}}$. The solver is not called when $d = 0$. When $d = 1$, it returns the unique feasible allocation $w = (1)$.

C.2.9 Detached outer-gradient convention

Let $w_{\mathcal{B}}^{\mathrm{FW}}$ be the final allocation returned by Algorithm 2. In the outer neural-network update, FairMT detaches this allocation:

$$\bar{w}_{\mathcal{B}} = \mathrm{sg}\left(w_{\mathcal{B}}^{\mathrm{FW}}\right).$$

The active-set AHFDA estimate used in the primal and dual updates is

$$\widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}\left(\bar{\Psi}_{\mathcal{B}}; \bar{w}_{\mathcal{B}}\right) = \bar{w}_{\mathcal{B}}^\top \bar{\Psi}_{\mathcal{B}} - \frac{\mu}{2}\|\bar{w}_{\mathcal{B}}\|_2^2 + \frac{\mu}{2M_{\mathcal{B}}}. \tag{38}$$

The allocation $\bar{w}_{\mathcal{B}}$ is detached. Hence the gradient with respect to the normalized active violation vector is

$$\nabla_{\bar{\Psi}_{\mathcal{B}}} \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}\left(\bar{\Psi}_{\mathcal{B}}; \bar{w}_{\mathcal{B}}\right) = \bar{w}_{\mathcal{B}}. \tag{39}$$

If the inner problem is solved exactly, then $\bar{w}_{\mathcal{B}} = w^\star(\bar{\Psi}_{\mathcal{B}})$, Eq. (38) equals the exact shifted AHFDA aggregate, and Eq. (39) recovers the Danskin gradient. With finite iterations, Eq. (39) is the gradient of the detached finite-step AHFDA estimate in Eq. (38).

Algorithm 2 Projection-Free AHFDA Solver

Require: Active normalized violation vector $v \in \mathbb{R}^d_{\geq 0}$, regularization $\mu > 0$, maximum iterations L_w, tolerance δ_{FW}
Ensure: Feasible allocation $w \in \Delta_d$
if $d = 0$ **then**
  Return empty allocation and skip the fairness update
end if
if $d = 1$ **then**
  Return $w = (1)$
end if
Initialize $j_{\mathrm{init}} \in \arg\max_j v_j$, $w^{(0)} = e_{j_{\mathrm{init}}}$
for $\ell = 0, \ldots, L_w - 1$ **do**
  Compute the gradient $r^{(\ell)} = \mu w^{(\ell)} - v$
  Select $j_\ell \in \arg\max_j (v_j - \mu w_j^{(\ell)})$, and set $s^{(\ell)} = e_{j_\ell}$
  Compute the Frank–Wolfe gap $\mathcal{G}_{\mathrm{FW}}^{(\ell)} = \langle r^{(\ell)}, w^{(\ell)} - s^{(\ell)} \rangle$
  if $\mathcal{G}_{\mathrm{FW}}^{(\ell)} \leq \delta_{\mathrm{FW}}$ **then**
    break
  end if
  Set $p^{(\ell)} = s^{(\ell)} - w^{(\ell)}$
  if $\|p^{(\ell)}\|_2 = 0$ **then**
    break
  end if
  Set $\gamma_\ell = \Pi_{[0,1]}\big(-\langle r^{(\ell)}, p^{(\ell)} \rangle / (\mu \|p^{(\ell)}\|_2^2)\big)$
  Update $w^{(\ell+1)} = w^{(\ell)} + \gamma_\ell p^{(\ell)}$
end for
return w

Since the normalized coordinate is

$$\bar{\psi}_{q_j} = \frac{\psi_{q_j}}{\nu_{q_j}}, \qquad q_j \in \mathcal{Q}_{\mathcal{B}},$$

the gradient passed to the raw coordinate ψ_{q_j} is

$$\frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial \psi_{q_j}} = \frac{\bar{w}_{\mathcal{B},j}}{\nu_{q_j}}.$$

Thus, coordinate-wise normalization affects both allocation and gradient scale. If $M_{\mathcal{B}} = 0$, no estimate in Eq. (38) is formed, the fairness-gradient contribution is skipped, and the dual multiplier is left unchanged for that update.

D Derivation of the Head-Aware Optimization Proxy

This appendix derives the Head-Aware Optimization Proxy used in Section 3.3. The goal is to replace the high-dimensional shared-parameter direction-finding problem in Eq. (7) with a low-dimensional task-space quadratic program. The proxy keeps the distinction between the two simplex variables used by FairMT: AHFDA computes allocation weights w over fairness coordinates, while the head-aware proxy computes task-combination coefficients α over utility tasks. The Head-Aware Optimization Proxy converts the coordinate-level AHFDA signal into task-level coefficients through three steps. First, it constructs utility directions a_t and fairness directions c_t in task-head representation space. Second, it approximates the constrained shared-gradient residual by the head-space residual in Eq. (45). Third, it solves the convex simplex QP in Eq. (46) to obtain the active-task weights used in the primal update. The proxy is therefore not a standalone fairness loss; it is a low-dimensional routing mechanism that maps normalized coordinate-level fairness pressure into task-combination weights for constrained multi-task optimization.

D.1 Active Tasks, Active Coordinates, and Head-Space Notation

For a mini-batch $\mathcal{B}$, the active utility-task set is

$$\mathcal{T}_{\mathcal{B}} = \left\{ t \in \mathcal{T} : \sum_{i \in \mathcal{B}} m_{i,t} > 0 \right\}, \qquad T_{\mathcal{B}} = |\mathcal{T}_{\mathcal{B}}|.$$

The head-aware QP is solved over $\mathcal{T}_{\mathcal{B}}$. After solving, the resulting active-task weights are expanded to the full task set by assigning zero weight to inactive tasks.

Let the active fairness-coordinate set be

$$\mathcal{Q}_{\mathcal{B}} = \{q \in \mathcal{Q} : q \text{ is active in mini-batch } \mathcal{B}\}, \qquad M_{\mathcal{B}} = |\mathcal{Q}_{\mathcal{B}}|.$$

If $M_{\mathcal{B}} = 0$, no valid fairness estimate is formed in the current mini-batch. In that case, all fairness head-space directions are set to zero and the head-aware QP reduces to a utility-only task-weighting problem.

For each input x_i, define the shared representation

$$r_i = f_{\theta_s}(x_i) \in \mathbb{R}^D.$$

For task t, let the task head produce an output or logit vector

$$z_{i,t} = h_{\theta_t}^{(t)}(r_i) \in \mathbb{R}^{d_t},$$

where $d_t = 1$ for binary detection and scalar regression, and $d_t = C_t$ for a C_t-class classification task. For each output dimension $k \in \{1, \ldots, d_t\}$, define the head Jacobian direction

$$H_{i,t,k} = \frac{\partial z_{i,t,k}}{\partial r_i} \in \mathbb{R}^D. \tag{40}$$

This notation covers both linear and nonlinear task heads. For a head affine in r_i, write $z_{i,t,k} = W_{t,k} r_i + b_{t,k}$, where $W_{t,k}$ is row k of the head weight matrix. The Jacobian direction is then the constant column $H_{t,k} = W_{t,k}^\top$.

D.2 Utility Head-Space Directions

The masked mini-batch task risk is $\widehat{L}_t^{\mathcal{B}}(\theta)$. Its gradient with respect to the shared representation of sample i can be written through the task head as

$$\nabla_{r_i} \widehat{L}_t^{\mathcal{B}}(\theta) = \sum_{k=1}^{d_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t,k}} H_{i,t,k}, \qquad i \in \mathcal{B}_t,$$

where

$$\mathcal{B}_t = \{i \in \mathcal{B} : m_{i,t} = 1\}.$$

The utility head-space direction for task t is defined as the aggregate representation-level direction induced by its head:

$$a_t = \sum_{i \in \mathcal{B}_t} \sum_{k=1}^{d_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t,k}} H_{i,t,k} \in \mathbb{R}^D. \tag{41}$$

This definition differentiates the masked mini-batch risk directly, retaining its loss-specific normalization and any cross-sample dependence. No additional batch averaging is applied when constructing a_t.

For scalar-output heads, $d_t = 1$, and Eq. (41) becomes

$$a_t = \sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t}} H_{i,t}.$$

If the scalar head is affine in r_i, its Jacobian direction is constant, $H_{i,t} = H_t$, giving

$$a_t = \left(\sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t}} \right) H_t.$$

For a multi-class classification task with C_t classes, $d_t = C_t$, and

$$a_t = \sum_{i \in \mathcal{B}_t} \sum_{c=1}^{C_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t,c}} H_{i,t,c}.$$

If each class logit is affine in r_i, then $H_{i,t,c} = H_{t,c}$ and

$$a_t = \sum_{c=1}^{C_t} \left(\sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}(\theta)}{\partial z_{i,t,c}} \right) H_{t,c}.$$

D.3 How AHFDA Enters the Fairness Head-Space Direction

When $M_{\mathcal{B}} > 0$, FairMT forms the normalized active violation vector

$$\bar{\Psi}_{\mathcal{B}}(\theta) = \left(\bar{\psi}_q(\theta)\right)_{q \in \mathcal{Q}_{\mathcal{B}}} = \left(\frac{\psi_q(\theta)}{\nu_q} \right)_{q \in \mathcal{Q}_{\mathcal{B}}},$$

and computes a detached AHFDA allocation $\bar{w}_{\mathcal{B}} \in \Delta_{M_{\mathcal{B}}}$. The active-set AHFDA estimate used in the outer update is

$$\widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}} \left(\bar{\Psi}_{\mathcal{B}}(\theta); \bar{w}_{\mathcal{B}} \right) = \bar{w}_{\mathcal{B}}^{\top} \bar{\Psi}_{\mathcal{B}}(\theta) - \frac{\mu}{2} \|\bar{w}_{\mathcal{B}}\|_2^2 + \frac{\mu}{2M_{\mathcal{B}}}.$$

The allocation $\bar{w}_{\mathcal{B}}$ is detached in the outer neural-network update, and its derivative is not backpropagated through the Frank–Wolfe solver. Therefore, for any logit $z_{i,t,k}$,

$$\frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t,k}} = \sum_{\substack{q \in \mathcal{Q}_{\mathcal{B}} \\ t(q)=t}} \bar{w}_q \frac{\partial \bar{\psi}_q(\theta)}{\partial z_{i,t,k}}.$$

Using $\bar{\psi}_q = \psi_q / \nu_q$, this becomes

$$\frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t,k}} = \sum_{\substack{q \in \mathcal{Q}_{\mathcal{B}} \\ t(q)=t}} \frac{\bar{w}_q}{\nu_q} \frac{\partial \psi_q(\theta)}{\partial z_{i,t,k}}. \tag{42}$$

Eq. (42) is the link between coordinate-level fairness allocation and task-head routing. The factor $\bar{w}_q$ determines how much AHFDA pressure coordinate q receives, while $1/\nu_q$ converts normalized-coordinate pressure back to the raw-coordinate scale.

The fairness head-space direction for task t is then defined as

$$c_t = \sum_{i \in \mathcal{B}_t} \sum_{k=1}^{d_t} \frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t,k}} H_{i,t,k} \in \mathbb{R}^D. \tag{43}$$

If $M_{\mathcal{B}} = 0$, no AHFDA estimate is formed, and FairMT sets

$$c_t = 0, \qquad \forall t \in \mathcal{T}_{\mathcal{B}}.$$

For scalar-output heads, Eq. (43) reduces to

$$c_t = \sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t}} H_{i,t}.$$

For a scalar head affine in r_i, $H_{i,t} = H_t$, giving

$$c_t = \left(\sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t}} \right) H_t.$$

For a multi-class classification head, the fairness direction becomes

$$c_t = \sum_{i \in \mathcal{B}_t} \sum_{c=1}^{C_t} \frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t,c}} H_{i,t,c}.$$

This expression naturally accounts for the one-vs-rest class-wise fairness coordinates induced by a multi-class classification utility head. Each class-wise coordinate contributes to the logit sensitivities through Eq. (42).

D.4 Head-Space Approximation to the Constrained Direction

The ideal constrained direction in shared-parameter space is motivated by the stationarity residual

$$\mathcal{R}_s(\alpha) = \frac{1}{2}\left\| \sum_{t\in\mathcal{T}_\mathcal{B}} \alpha_t \nabla_{\theta_s} \widehat{L}_t^{\mathcal{B}}(\theta) + \lambda \nabla_{\theta_s} \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}(\theta) \right\|_2^2. \tag{44}$$

Let

$$J_i = \frac{\partial r_i}{\partial \theta_s}$$

be the shared-representation Jacobian. The shared-parameter utility gradient can be decomposed as

$$\nabla_{\theta_s} \widehat{L}_t^{\mathcal{B}}(\theta) = \sum_{i\in\mathcal{B}_t} J_i^\top \nabla_{r_i} \widehat{L}_t^{\mathcal{B}}(\theta).$$

Similarly, the shared-parameter AHFDA gradient decomposes as

$$\nabla_{\theta_s} \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}(\theta) = \sum_{i\in\mathcal{B}} J_i^\top \nabla_{r_i} \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}(\theta).$$

The proxy routes utility and fairness gradients through task t's head, then sums the resulting directions across samples to form a_t and c_t in representation space. This sum omits the sample-specific encoder Jacobians J_i. A sum that vanishes in representation space can become nonzero when each contribution is first multiplied by $J_i^\top$. The two residuals can therefore have different minimizers. The aggregate head-space fairness direction is

$$g_\Phi^h = \sum_{t\in\mathcal{T}_\mathcal{B}} c_t.$$

Thus, the head-space analogue of Eq. (44) is

$$\mathcal{R}_h(\alpha) = \frac{1}{2}\left\| \sum_{t\in\mathcal{T}_\mathcal{B}} \alpha_t a_t + \lambda \sum_{t\in\mathcal{T}_\mathcal{B}} c_t \right\|_2^2. \tag{45}$$

The plus sign follows the Lagrangian stationarity condition. The actual model update later descends the local primal objective and takes a step in the negative gradient direction.

D.5 Convex Task-Space Quadratic Program

Stack the active utility directions row-wise into

$$A_\mathcal{B} = \begin{bmatrix} a_{t_1}^\top \\ \vdots \\ a_{t_{T_\mathcal{B}}}^\top \end{bmatrix} \in \mathbb{R}^{T_\mathcal{B}\times D}, \qquad t_j \in \mathcal{T}_\mathcal{B}.$$

Similarly, stack the active fairness directions row-wise into

$$C_\mathcal{B} = \begin{bmatrix} c_{t_1}^\top \\ \vdots \\ c_{t_{T_\mathcal{B}}}^\top \end{bmatrix} \in \mathbb{R}^{T_\mathcal{B}\times D}.$$

Then Eq. (45) can be written as

$$\mathcal{R}_h(\alpha) = \frac{1}{2}\left\| A_\mathcal{B}^\top \alpha + \lambda C_\mathcal{B}^\top \mathbf{1}_{T_\mathcal{B}} \right\|_2^2, \qquad \alpha \in \Delta_{T_\mathcal{B}}.$$

Expanding the squared norm gives

$$\mathcal{R}_h(\alpha) = \frac{1}{2}\alpha^\top A_\mathcal{B} A_\mathcal{B}^\top \alpha + \lambda \alpha^\top A_\mathcal{B} C_\mathcal{B}^\top \mathbf{1}_{T_\mathcal{B}} + \frac{\lambda^2}{2}\left\| C_\mathcal{B}^\top \mathbf{1}_{T_\mathcal{B}} \right\|_2^2.$$

The last term is independent of α and can be dropped from the task-weight optimization. Define

$$K_\mathcal{B} = A_\mathcal{B} A_\mathcal{B}^\top, \qquad b_\mathcal{B} = \lambda A_\mathcal{B} C_\mathcal{B}^\top \mathbf{1}_{T_\mathcal{B}}.$$

The head-aware task-weighting problem is therefore

$$\alpha_{\mathcal{B}} = \arg\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{1}{2}\alpha^\top K_{\mathcal{B}}\alpha + b_{\mathcal{B}}^\top \alpha. \tag{46}$$

This is the active-task version of Eq. (8) in the main text. For readability, the main text suppresses the mini-batch subscript and writes $K = AA^\top$ and $b = \lambda AC^\top \mathbf{1}$.

Proposition 5 (Convexity of the head-aware QP)**.** The optimization problem in Eq. (46) is convex over the simplex $\Delta_{T_{\mathcal{B}}}$.

Proof. For any $v \in \mathbb{R}^{T_{\mathcal{B}}}$,

$$v^\top K_{\mathcal{B}} v = v^\top A_{\mathcal{B}} A_{\mathcal{B}}^\top v = \left\| A_{\mathcal{B}}^\top v \right\|_2^2 \geq 0.$$

Therefore, $K_{\mathcal{B}} \succeq 0$. The objective in Eq. (46) is a convex quadratic function plus a linear term, and $\Delta_{T_{\mathcal{B}}}$ is a compact convex set. Hence the problem is convex and admits at least one global minimizer. □

If $M_{\mathcal{B}} = 0$, then $C_{\mathcal{B}} = 0$ and $b_{\mathcal{B}} = 0$. If the dual multiplier is inactive, $\lambda = 0$, then $b_{\mathcal{B}} = 0$ as well. In both cases, Eq. (46) reduces to a utility-only head-aware task-weighting problem.

D.6 Projection-Free Solver for the Head-Aware QP

The QP in Eq. (46) is solved with a projection-free Frank–Wolfe procedure over the active-task simplex. Define

$$f_{\text{head}}(\alpha) = \frac{1}{2}\alpha^\top K_{\mathcal{B}}\alpha + b_{\mathcal{B}}^\top \alpha.$$

Its gradient is

$$\nabla f_{\text{head}}(\alpha) = K_{\mathcal{B}}\alpha + b_{\mathcal{B}}.$$

At inner iteration ℓ, the linear minimization oracle is

$$s^{(\ell)} \in \arg\min_{s \in \Delta_{T_{\mathcal{B}}}} \left\langle K_{\mathcal{B}}\alpha^{(\ell)} + b_{\mathcal{B}}, s \right\rangle.$$

Since the feasible set is a simplex, the solution is a vertex:

$$s^{(\ell)} = e_{j_\ell}, \qquad j_\ell \in \arg\min_{j \in \{1,\dots,T_{\mathcal{B}}\}} \left(K_{\mathcal{B}}\alpha^{(\ell)} + b_{\mathcal{B}} \right)_j.$$

The Frank–Wolfe direction is

$$d^{(\ell)} = s^{(\ell)} - \alpha^{(\ell)}.$$

If $(d^{(\ell)})^\top K_{\mathcal{B}} d^{(\ell)} > 0$, exact line search gives

$$\gamma_\ell = \Pi_{[0,1]} \left(-\frac{\left\langle K_{\mathcal{B}}\alpha^{(\ell)} + b_{\mathcal{B}}, d^{(\ell)} \right\rangle}{(d^{(\ell)})^\top K_{\mathcal{B}} d^{(\ell)}} \right).$$

If $(d^{(\ell)})^\top K_{\mathcal{B}} d^{(\ell)} = 0$, the line objective is linear. In that case, we set

$$\gamma_\ell = \begin{cases} 1, & \left\langle K_{\mathcal{B}}\alpha^{(\ell)} + b_{\mathcal{B}}, d^{(\ell)} \right\rangle < 0, \\ 0, & \left\langle K_{\mathcal{B}}\alpha^{(\ell)} + b_{\mathcal{B}}, d^{(\ell)} \right\rangle \geq 0. \end{cases}$$

The update is

$$\alpha^{(\ell+1)} = \alpha^{(\ell)} + \gamma_\ell d^{(\ell)} = (1 - \gamma_\ell)\alpha^{(\ell)} + \gamma_\ell s^{(\ell)}.$$

Since both $\alpha^{(\ell)}$ and $s^{(\ell)}$ belong to the simplex and $\gamma_\ell \in [0, 1]$, all iterates remain feasible:

$$\alpha^{(\ell)} \in \Delta_{T_{\mathcal{B}}}, \qquad \forall \ell \geq 0.$$

The Frank–Wolfe gap provides an optimality certificate:

$$G_\ell^\alpha = \left\langle \nabla f_{\text{head}}(\alpha^{(\ell)}), \alpha^{(\ell)} - s^{(\ell)} \right\rangle.$$

For any minimizer $\alpha^\star$ of Eq. (46),

$$f_{\mathrm{head}}(\alpha^{(\ell)}) - f_{\mathrm{head}}(\alpha^\star) \le G_\ell^\alpha.$$

Moreover, since f_{head} is smooth with Hessian $K_{\mathcal{B}}$, the standard Frank–Wolfe curvature bound gives, for $\ell \ge 1$,

$$f_{\mathrm{head}}(\alpha^{(\ell)}) - f_{\mathrm{head}}(\alpha^\star) \le \frac{4\|K_{\mathcal{B}}\|_2}{\ell + 2},$$

where we used the Euclidean diameter of the probability simplex, $\mathrm{diam}(\Delta_{T_{\mathcal{B}}}) = \sqrt{2}$. In practice, FairMT uses a small fixed number of Frank–Wolfe iterations L_α, optionally with early stopping when $G_\ell^\alpha \le \delta_{\mathrm{FW}}$.

Let ℓ_{final} be the index of the final solver iterate. The outer update uses the detached task weights

$$\bar{\alpha}_{\mathcal{B}} = \mathrm{sg}\left(\alpha^{(\ell_{\mathrm{final}})}\right).$$

Thus, FairMT does not backpropagate through the QP solver itself. The weights determine how task losses are combined in the current primal update, but they are not model parameters.

D.7 Relation to MGDA-Style Weighting

When no fairness coordinate is active or when the dual multiplier is zero, $b_{\mathcal{B}} = 0$, and the head-aware QP becomes

$$\alpha_{\mathcal{B}} = \arg\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{1}{2}\alpha^\top A_{\mathcal{B}} A_{\mathcal{B}}^\top \alpha = \arg\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{1}{2}\left\|A_{\mathcal{B}}^\top \alpha\right\|_2^2. \tag{47}$$

This is a minimum-norm task-combination problem in head space. If the head-space directions a_t coincide with the true shared-parameter gradients $\nabla_{\theta_s} \widehat{L}_t^{\mathcal{B}}$, Eq. (47) recovers the MGDA-style utility-only direction-finding problem. In the general case, the proxy compares tasks through output-head geometry and output sensitivities, while full MGDA uses shared-parameter gradients.

A further special case illustrates the behavior of the proxy. If the effective utility directions are orthogonal up to a common scale,

$$A_{\mathcal{B}} A_{\mathcal{B}}^\top = c I_{T_{\mathcal{B}}}, \qquad c > 0,$$

and $b_{\mathcal{B}} = 0$, then Eq. (47) becomes

$$\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{c}{2}\|\alpha\|_2^2.$$

Its minimizer is the uniform active-task weighting,

$$\alpha_t^\star = \frac{1}{T_{\mathcal{B}}}, \qquad t \in \mathcal{T}_{\mathcal{B}}.$$

When fairness is active, the linear term $b_{\mathcal{B}}$ shifts this utility-only weighting toward task combinations that better align with the routed AHFDA constraint signal.

E FairMT Optimization and Training Procedure

This appendix expands the fairness-constrained optimization procedure used by FairMT. The main text presents the population-level problem and summarizes the primal–dual loop in Algorithm 1. Algorithm 3 gives the full mini-batch procedure. Here we make explicit the stochastic active-set surrogate, the role of stop-gradient references, the constrained Pareto-stationarity motivation, the fairness-adjusted direction-finding problem, and the exact mini-batch training procedure. Throughout this section, AHFDA allocation weights w are defined over fairness coordinates, while task-combination coefficients α are defined over utility tasks.

E.1 Local Detached-Reference Constrained Surrogate

FairMT uses asymmetric coordinates whose reference statistics are detached. At each iteration, the algorithm optimizes the local surrogate with the reference statistics fixed. For a mini-batch $\mathcal{B}$, the active utility-task set is

$$\mathcal{T}_{\mathcal{B}} = \left\{ t \in \mathcal{T} : \sum_{i \in \mathcal{B}} m_{i,t} > 0 \right\}, \qquad T_{\mathcal{B}} = |\mathcal{T}_{\mathcal{B}}|. \tag{48}$$

For each active task with a decomposable loss, the masked empirical task loss is

$$\widehat{L}_t^{\mathcal{B}}(\theta) = \frac{\sum_{i \in \mathcal{B}} m_{i,t} \ell_t(\hat{y}_{i,t}, y_{i,t})}{\max(1, \sum_{i \in \mathcal{B}} m_{i,t})}, \qquad t \in \mathcal{T}_{\mathcal{B}}. \tag{49}$$

For regression, use the batch loss in Eq. (74) and the small-batch case in Eq. (75).

For the same mini-batch, let $\mathcal{Q}_{\mathcal{B}} \subseteq \mathcal{Q}$ be the active fairness-coordinate set whose required group comparisons are observable. Its size is

$$M_{\mathcal{B}} = |\mathcal{Q}_{\mathcal{B}}|.$$

The active-set convention follows Appendix A: a discrete coordinate is active when $Z_{q,\mathcal{B}} > 0$ in Eq. (10), and a regression coordinate is active when at least two valid groups are observed.

At iteration r, the model parameters are $\theta^{(r)}$. For each active coordinate $q \in \mathcal{Q}_{\mathcal{B}}$, the detached-reference coordinate $\psi_{q,\mathcal{B}}^{(r,\mathrm{sg})}(\theta)$ is defined by the same asymmetric coordinate formula as in Appendix B, except that the reference statistic is computed at the current forward pass and treated as a constant in the backward pass. For each valid discrete component, the corresponding reference is

$$\tau_{q,\mathcal{B}}^{(r)} = \mathrm{sg}\left(\max_{g \in \mathcal{G}_{q,\mathcal{B}}^{+}} \widehat{\mathrm{TPR}}_{g,q}^{\mathcal{B}}(\theta^{(r)}) \right), \qquad \phi_{q,\mathcal{B}}^{(r)} = \mathrm{sg}\left(\min_{g \in \mathcal{G}_{q,\mathcal{B}}^{-}} \widehat{\mathrm{FPR}}_{g,q}^{\mathcal{B}}(\theta^{(r)}) \right).$$

For regression coordinates, this means

$$e_{q,\mathcal{B}}^{\mathrm{ref},(r)} = \mathrm{sg}\left(\min_{g \in \mathcal{G}_{q,\mathcal{B}}^{\mathrm{reg}}} \widehat{e}_{g,q}^{\mathcal{B}}(\theta^{(r)}) \right).$$

The raw active detached-reference violation vector is

$$\Psi_{\mathcal{B},\mathrm{sg}}^{(r)}(\theta) = \left(\psi_{q,\mathcal{B}}^{(r,\mathrm{sg})}(\theta) \right)_{q \in \mathcal{Q}_{\mathcal{B}}} \in \mathbb{R}_{\geq 0}^{M_{\mathcal{B}}}. \tag{50}$$

The normalized active vector is

$$\bar{\Psi}_{\mathcal{B},\mathrm{sg}}^{(r)}(\theta) = \left(\bar{\psi}_{q,\mathcal{B}}^{(r,\mathrm{sg})}(\theta) \right)_{q \in \mathcal{Q}_{\mathcal{B}}} = \left(\frac{\psi_{q,\mathcal{B}}^{(r,\mathrm{sg})}(\theta)}{\nu_q} \right)_{q \in \mathcal{Q}_{\mathcal{B}}}, \tag{51}$$

where $\nu_q > 0$ is the fixed validation-calibrated coordinate scale from Appendix A. AHFDA is always applied to the normalized active vector in Eq. (51), not to the raw vector in Eq. (50).

When $M_{\mathcal{B}} > 0$, FairMT computes an AHFDA allocation on the current normalized active vector and detaches it:

$$\bar{w}_{\mathcal{B}}^{(r)} = \mathrm{sg}\left(\text{FW-AHFDA}\left(\bar{\Psi}_{\mathcal{B},\mathrm{sg}}^{(r)}(\theta^{(r)}); \mu \right) \right), \qquad \bar{w}_{\mathcal{B}}^{(r)} \in \Delta_{M_{\mathcal{B}}}. \tag{52}$$

Using this detached allocation, the active-set AHFDA estimate used in the outer neural-network update is

$$\widehat{\Phi}_{\mu,\mathcal{B}}^{(r)}(\theta) = \left(\bar{w}_{\mathcal{B}}^{(r)} \right)^{\top} \bar{\Psi}_{\mathcal{B},\mathrm{sg}}^{(r)}(\theta) - \frac{\mu}{2} \left\| \bar{w}_{\mathcal{B}}^{(r)} \right\|_2^2 + \frac{\mu}{2M_{\mathcal{B}}}. \tag{53}$$

The corresponding local constraint residual is

$$h_{\mathcal{B}}^{(r)}(\theta) = \widehat{\Phi}_{\mu,\mathcal{B}}^{(r)}(\theta) - \epsilon. \tag{54}$$

The scalar residual used for the dual update at iteration r is evaluated before the primal update:

$$h_{\mathcal{B}}^{(r)} = h_{\mathcal{B}}^{(r)}(\theta^{(r)}) = \widehat{\Phi}_{\mu,\mathcal{B}}^{(r)}(\theta^{(r)}) - \epsilon. \tag{55}$$

If $M_{\mathcal{B}} = 0$, the mini-batch contains no valid fairness-coordinate comparison. In that case, no AHFDA allocation is formed, no fairness residual is evaluated, the primal update uses only utility losses, and the dual multiplier is left unchanged.

The main text writes the constraint compactly as

$$\widetilde{\Phi}_{\mu}(\bar{\Psi}(\theta)) \leq \epsilon. \tag{56}$$

In stochastic training, Eq. (56) is implemented by the active-set detached-reference estimate in Eq. (53) and the residual in Eq. (54).

E.2 Fairness-Constrained Pareto Stationarity

The first-order stationarity condition for FairMT's local detached-reference surrogate motivates its direction-finding problem. For readability, consider the full-coordinate version of the detached-reference surrogate. Let

$$\Psi_{\mathrm{sg}}(\theta) = \left(\psi_q^{\mathrm{sg}}(\theta)\right)_{q\in\mathcal{Q}}$$

be the raw detached-reference violation vector, and let

$$\bar{\Psi}_{\mathrm{sg}}(\theta) = \left(\frac{\psi_q^{\mathrm{sg}}(\theta)}{\nu_q}\right)_{q\in\mathcal{Q}}$$

be its normalized version. The full-coordinate surrogate constraint residual is

$$h_{\mathrm{sg}}(\theta) = \widetilde{\Phi}_{\mu}\left(\bar{\Psi}_{\mathrm{sg}}(\theta)\right) - \epsilon.$$

Definition 1 (Fairness-constrained Pareto stationarity for the local surrogate)**.** A feasible point $\theta^{\star}$ is fairness-constrained Pareto-stationary for the local detached-reference surrogate if there exist task-combination coefficients $\alpha^{\star} \in \Delta_T$ and a multiplier $\lambda^{\star} \geq 0$ such that primal feasibility, complementary slackness, and first-order stationarity hold:

$$h_{\mathrm{sg}}(\theta^{\star}) \leq 0, \qquad \lambda^{\star} h_{\mathrm{sg}}(\theta^{\star}) = 0, \qquad \lambda^{\star} \geq 0,$$

and

$$\sum_{t=1}^{T} \alpha_t^{\star} \nabla_{\theta} L_t(\theta^{\star}) + \lambda^{\star} \nabla_{\theta} \widetilde{\Phi}_{\mu}\left(\bar{\Psi}_{\mathrm{sg}}(\theta^{\star})\right) = 0. \tag{57}$$

When max or hinge operators are not smoothed, Eq. (57) should be interpreted using a valid selected subgradient of the local surrogate. Away from tie points, the ordinary gradient is well-defined. In implementation, automatic differentiation selects one such subgradient for the active max and hinge branches.

Proposition 6 (First-order certificate for the local FairMT surrogate)**.** Assume that, for a fixed detached-reference configuration, the local surrogate objectives $L_t(\theta)$ and $h_{\mathrm{sg}}(\theta)$ are continuously differentiable. If smooth relaxations are not used, replace gradients by valid selected subgradients. If a standard constraint qualification and a non-abnormality condition hold at a local constrained Pareto optimum $\theta^{\star}$ of the detached-reference surrogate problem, then there exist $\alpha^{\star} \in \Delta_T$ and $\lambda^{\star} \geq 0$ satisfying Definition 1.

Proof. At a local constrained Pareto optimum of the detached-reference surrogate problem, there is no feasible first-order direction that simultaneously decreases all task objectives while remaining in the feasible set. The feasible set is defined by

$$h_{\mathrm{sg}}(\theta) \leq 0.$$

Under a standard constraint qualification, the separating-hyperplane argument for constrained multi-objective optimization gives nonnegative objective multipliers $\tilde{\alpha}_1, \ldots, \tilde{\alpha}_T$ and a nonnegative constraint multiplier $\tilde{\lambda} \geq 0$, not all zero, such that

$$\sum_{t=1}^{T} \tilde{\alpha}_t \nabla_{\theta} L_t(\theta^{\star}) + \tilde{\lambda} \nabla_{\theta} h_{\mathrm{sg}}(\theta^{\star}) = 0. \tag{58}$$

The non-abnormality condition ensures that the objective multipliers are not all zero. Therefore,

$$S = \sum_{t=1}^{T} \tilde{\alpha}_t > 0.$$

Define normalized task multipliers and the normalized constraint multiplier as

$$\alpha_t^\star = \frac{\tilde{\alpha}_t}{S}, \qquad \lambda^\star = \frac{\tilde{\lambda}}{S}.$$

Then $\alpha^\star \in \Delta_T$ and $\lambda^\star \geq 0$. Since

$$h_{\mathrm{sg}}(\theta) = \widetilde{\Phi}_\mu\left(\bar{\Psi}_{\mathrm{sg}}(\theta)\right) - \epsilon,$$

we have

$$\nabla_\theta h_{\mathrm{sg}}(\theta^\star) = \nabla_\theta \widetilde{\Phi}_\mu\left(\bar{\Psi}_{\mathrm{sg}}(\theta^\star)\right).$$

Dividing Eq. (58) by S gives the stationarity condition in Eq. (57). Primal feasibility, dual feasibility, and complementary slackness follow from the KKT conditions for the inequality constraint. □

Remark 1. Proposition 6 justifies the local stationarity residual under a fixed detached-reference surrogate. The stop-gradient references specify the surrogate gradient used by the algorithm: reference statistics affect the forward violation value, while explicit fairness-gradient pressure is applied through currently violating group statistics. AHFDA is applied to the normalized surrogate vector $\bar{\Psi}_{\mathrm{sg}}$. The gradient with respect to a raw coordinate ψ_q therefore includes the fixed factor $1/\nu_q$.

E.3 Fairness-Adjusted Direction Finding

At a fairness-constrained Pareto-stationary point of the local surrogate, the shared-parameter block satisfies

$$\sum_{t=1}^{T} \alpha_t^\star \nabla_{\theta_s} L_t(\theta^\star) + \lambda^\star \nabla_{\theta_s} \widetilde{\Phi}_\mu\left(\bar{\Psi}_{\mathrm{sg}}(\theta^\star)\right) = 0.$$

FairMT uses this condition to construct a local task-combination problem at every stochastic update. For each active utility task $t \in \mathcal{T}_\mathcal{B}$, define the shared-parameter utility gradient at iteration r as

$$g_{t,\mathcal{B}}^{s,(r)} = \nabla_{\theta_s} \widehat{L}_t^{\mathcal{B}}(\theta^{(r)}).$$

When $M_\mathcal{B} > 0$, define the shared-parameter fairness gradient as

$$g_{\Phi,\mathcal{B}}^{s,(r)} = \nabla_{\theta_s} \widehat{\Phi}_{\mu,\mathcal{B}}^{(r)}(\theta^{(r)}).$$

When $M_\mathcal{B} = 0$, no fairness-gradient term is available and we set the fairness term to zero for direction finding.

For $M_\mathcal{B} > 0$, the ideal task weights minimize the local constrained-stationarity residual

$$\alpha_\mathcal{B}^{(r)} = \arg\min_{\alpha \in \Delta_{T_\mathcal{B}}} \frac{1}{2} \left\| \sum_{t \in \mathcal{T}_\mathcal{B}} \alpha_t g_{t,\mathcal{B}}^{s,(r)} + \lambda^{(r)} g_{\Phi,\mathcal{B}}^{s,(r)} \right\|_2^2. \tag{59}$$

If $M_\mathcal{B} = 0$, the direction-finding problem reduces to the utility-only minimum-norm task-combination problem:

$$\alpha_\mathcal{B}^{(r)} = \arg\min_{\alpha \in \Delta_{T_\mathcal{B}}} \frac{1}{2} \left\| \sum_{t \in \mathcal{T}_\mathcal{B}} \alpha_t g_{t,\mathcal{B}}^{s,(r)} \right\|_2^2.$$

When $M_\mathcal{B} > 0$ and $\lambda^{(r)} = 0$, Eq. (59) also reduces to the utility-only direction.

The exact shared-gradient problem in Eq. (59) can be expensive for large neural networks. FairMT therefore computes the task weights for the primal update with the Head-Aware Optimization Proxy in Appendix D. The proxy solves a low-dimensional task-space QP over the active utility tasks

$$\alpha_\mathcal{B}^{(r)} = \arg\min_{\alpha \in \Delta_{T_\mathcal{B}}} \frac{1}{2} \alpha^\top K_\mathcal{B}^{(r)} \alpha + \left(b_\mathcal{B}^{(r)}\right)^\top \alpha. \tag{60}$$

Here $K_{\mathcal{B}}^{(r)}$ is the active-task head-space Gram matrix and $b_{\mathcal{B}}^{(r)}$ is the fairness-induced linear term. If no fairness coordinate is active, then $b_{\mathcal{B}}^{(r)} = 0$, and the proxy becomes utility-only task weighting.

After solving Eq. (60), FairMT expands the active-task coefficient vector to the full task set by assigning zero weight to inactive tasks:

$$\alpha_t^{(r)} = \begin{cases} \left(\alpha_{\mathcal{B}}^{(r)}\right)_t, & t \in \mathcal{T}_{\mathcal{B}}, \\ 0, & t \notin \mathcal{T}_{\mathcal{B}}. \end{cases} \tag{61}$$

The expanded coefficient is treated as a local optimization coefficient, not as a model parameter. In the outer neural-network update, FairMT uses the detached coefficient

$$\bar{\alpha}^{(r)} = \mathrm{sg}\left(\alpha^{(r)}\right). \tag{62}$$

E.4 Stochastic Primal–Dual Training

Given the detached AHFDA allocation $\bar{w}_{\mathcal{B}}^{(r)}$ and detached task weights $\bar{\alpha}^{(r)}$, FairMT forms a local primal objective. If $M_{\mathcal{B}} > 0$, the objective is

$$\mathcal{J}_{\mathcal{B}}^{(r)}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \bar{\alpha}_t^{(r)} \widehat{L}_t^{\mathcal{B}}(\theta) + \lambda^{(r)} \left[\widehat{\Phi}_{\mu,\mathcal{B}}^{(r)}(\theta) - \epsilon\right]. \tag{63}$$

If $M_{\mathcal{B}} = 0$, the fairness term is skipped and the local primal objective is

$$\mathcal{J}_{\mathcal{B}}^{(r)}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \bar{\alpha}_t^{(r)} \widehat{L}_t^{\mathcal{B}}(\theta). \tag{64}$$

The shared encoder is updated by

$$\theta_s^{(r+1)} = \theta_s^{(r)} - \rho_s \nabla_{\theta_s} \mathcal{J}_{\mathcal{B}}^{(r)}(\theta^{(r)}). \tag{65}$$

Each task head is updated by

$$\theta_t^{(r+1)} = \theta_t^{(r)} - \rho_t \nabla_{\theta_t} \mathcal{J}_{\mathcal{B}}^{(r)}(\theta^{(r)}), \qquad t \in \mathcal{T}. \tag{66}$$

Inactive task heads receive neither utility nor fairness gradients from the mini-batch objective, since each fairness coordinate inherits its parent task's observation mask.

When $M_{\mathcal{B}} > 0$, the dual multiplier is updated by projected ascent using the pre-update residual from Eq. (55):

$$\lambda^{(r+1)} = \left[\lambda^{(r)} + \rho_\lambda h_{\mathcal{B}}^{(r)}\right]_+. \tag{67}$$

When $M_{\mathcal{B}} = 0$, no valid fairness residual exists and the multiplier is left unchanged:

$$\lambda^{(r+1)} = \lambda^{(r)}. \tag{68}$$

Algorithm 3 summarizes the full stochastic procedure. The two inner simplex problems have different roles: AHFDA computes w over active fairness coordinates, while the head-aware proxy computes α over active utility tasks. Both are local optimization coefficients and are detached in the outer model-parameter update.

At inference time, FairMT uses only the trained shared encoder and task-specific heads. AHFDA allocation, the head-aware QP, active-coordinate construction, and the dual update are training-time components and do not introduce additional inference-time modules.

F Experimental Details

F.1 Datasets and Task Construction

This section describes the benchmark construction used in our experiments. We evaluate FairMT on three multi-task benchmarks spanning visual and textual modalities, homogeneous and heterogeneous

Algorithm 3 FairMT Stochastic Primal–Dual Training

Require: Training data $\mathcal{D}$, utility tasks $\mathcal{T}$, fairness coordinates $\mathcal{Q}$, coordinate scales $\{\nu_q\}_{q\in\mathcal{Q}}$, tolerance ϵ, AHFDA regularization μ, learning rates $\rho_s, \rho_t, \rho_\lambda$
Ensure: Trained parameters $\theta = (\theta_s, \{\theta_t\}_{t=1}^T)$

Initialize model parameters $\theta^{(0)}$ and multiplier $\lambda^{(0)} \geq 0$
for each training iteration $r = 0, 1, \ldots$ **do**
  Sample mini-batch $\mathcal{B}$
  Compute active utility-task set $\mathcal{T}_\mathcal{B}$ using Eq. (48)
  Compute masked losses using Eq. (49) for discrete tasks and Appendix F.4 for regression
  Compute active fairness-coordinate set $\mathcal{Q}_\mathcal{B}$ and $M_\mathcal{B}$
  if $M_\mathcal{B} > 0$ **then**
    Form normalized detached-reference vector using Eq. (51)
    Compute detached AHFDA allocation using Eq. (52)
    Compute local residual using Eq. (55)
    Solve the active-task head-aware QP in Eq. (60)
    Detach and expand task weights using Eqs. (61)–(62)
    Form the primal objective in Eq. (63)
  else
    Set $c_t = 0$ and solve Eq. (60) with $b_\mathcal{B}^{(r)} = 0$
    Detach and expand task weights using Eqs. (61)–(62)
    Form the utility-only primal objective in Eq. (64)
  end if
  Update θ_s using Eq. (65)
  Update all task heads using Eq. (66)
  if $M_\mathcal{B} > 0$ **then**
    Update λ using Eq. (67)
  else
    Keep λ unchanged using Eq. (68)
  end if
end for
return θ

task structures, full and partial supervision, and three output types: binary detection, multi-class classification, and scalar regression. Throughout this appendix, utility tasks and fairness coordinates are kept distinct. Utility is optimized over task heads $t \in \mathcal{T}$, whereas fairness is measured over output-side coordinates $q \in \mathcal{Q}$. A binary detection task contributes one fairness coordinate, a scalar regression target contributes one fairness coordinate, and a multi-class classification task contributes one one-vs-rest fairness coordinate per class.

For sample i, let $\Omega_i \subseteq \mathcal{T}$ denote the set of utility tasks with observed labels, and let $m_{i,t} = \mathbb{1}[t \in \Omega_i]$ be the corresponding observation mask. The same mask is used for both utility optimization and fairness-coordinate estimation. Thus, if $m_{i,t} = 0$, sample i contributes neither to the utility loss of task t nor to any fairness coordinate whose parent task is t. For a fairness coordinate q, we denote its parent utility task by $t(q)$ and use $m_{i,q} = m_{i,t(q)}$. For each fairness evaluation, one grouping map $s_i = \pi(a_i) \in \mathcal{G}$ is fixed, where a_i contains group-related metadata and $\mathcal{G}$ is the group set for the selected sensitive grouping. Changing the sensitive attribute changes π, but does not change the output-side coordinate construction.

F.1.1 Attribute Detection

We use CelebA [37] for homogeneous multi-task facial attribute detection. CelebA contains over 200K face images with binary facial-attribute annotations. We construct a 29-task benchmark in which each retained attribute is treated as one binary utility task. Since all retained outputs are binary detection tasks, each utility task also induces one fairness coordinate. Thus, Attribute Detection contains 29 utility tasks and 29 fairness coordinates. This benchmark is fully supervised: each sample has labels for all retained attribute tasks. We retain attributes that provide meaningful group-wise comparisons under the evaluated sensitive groupings. Attributes with extreme group-label imbalance

are excluded, because their positive or negative cells can be too concentrated in one group to support stable fairness estimation. The final 29 attribute tasks are:

> *Arched Eyebrows*, *Attractive*, *Bags Under Eyes*, *Bangs*, *Big Lips*, *Big Nose*, *Black Hair*, *Blond Hair*, *Blurry*, *Bushy Eyebrows*, *Brown Hair*, *Chubby*, *Double Chin*, *Eyeglasses*, *Gray Hair*, *High Cheekbones*, *Mouth Slightly Open*, *Narrow Eyes*, *Oval Face*, *Pale Skin*, *Pointy Nose*, *Receding Hairline*, *Smiling*, *Straight Hair*, *Wavy Hair*, *Wearing Earrings*, *Wearing Hat*, *Wearing Lipstick*, and *Wearing Necklace*.

Sensitive groupings. We evaluate fairness with respect to Age and Gender. Age is constructed from the provided *Young* metadata and uses the groups $\{\textit{Young}, \textit{Old}\}$. Gender is constructed from the provided *Male* indicator and uses the groups $\{\textit{Male}, \textit{Female}\}$. For each evaluation, a single group map $s_i = \pi(a_i)$ is fixed and shared across all 29 attribute coordinates.

Preprocessing and splits. We use the official CelebA train/validation/test partition. Images are aligned and center-cropped following the standard CelebA protocol, resized to 112×112, and normalized using the common visual preprocessing pipeline described in Appendix F.5. Since Attribute Detection has homogeneous binary outputs and full labels, it serves as the controlled visual benchmark for evaluating fairness-aware MTL without output-type heterogeneity or missing-task labels.

F.1.2 Affective Analysis

The Affective Analysis benchmark combines AffectNet [42], AffectNet-VA, and EmotioNet [15]. AffectNet provides expression labels, AffectNet-VA provides continuous valence–arousal labels, and EmotioNet provides action-unit annotations. This benchmark is heterogeneous because the shared model jointly learns multi-class classification, scalar regression, and binary detection. It is also partially supervised because the three datasets annotate different affective targets.

Utility tasks and fairness coordinates. The benchmark contains 11 utility tasks: one 7-class expression classification task, two scalar regression tasks for Valence and Arousal, and eight binary AU detection tasks. The corresponding fairness-coordinate set contains 17 coordinates: seven one-vs-rest expression coordinates, two regression coordinates, and eight AU coordinates.

The expression task uses seven classes:

> *Anger*, *Disgust*, *Fear*, *Happiness*, *Neutral*, *Sadness*, and *Surprise*.

The AU detection tasks are:

> AU1, AU2, AU4, AU5, AU6, AU12, AU25, and AU26.

Demographic annotation. AffectNet is a large-scale facial expression dataset collected from the web. We use Age, Gender, and Race annotations from prior affective fairness protocols [21, 23]. AffectNet-VA provides Valence and Arousal annotations in $[-1, 1]$ and uses the same demographic grouping protocol as AffectNet. We align EmotioNet demographic annotations with both AffectNet protocols so that all affective tasks use the same grouping map for a given evaluation.

Fairness is evaluated under Age, Gender, and Race. Age uses seven groups: $\{00\text{–}09, 10\text{–}19, 20\text{–}29, 30\text{–}39, 40\text{–}49, 50\text{–}59, 60+\}$. Gender uses $\{\textit{Male}, \textit{Female}\}$. Race uses $\{\textit{Asian}, \textit{Black}, \textit{Indian}, \textit{White}\}$. The source demographic annotations use slightly different age-bin definitions across datasets, so we merge fine-grained bins into the seven age groups above. For example, bins such as "< 2" and "03–09" are merged into the shared "00–09" group.

Partial supervision. We merge samples from AffectNet, AffectNet-VA, and EmotioNet into one heterogeneous benchmark. Since each dataset annotates different affective targets, a sample contributes labels only for its observed task subset Ω_i. A fairness coordinate q is considered only when its parent task $t(q)$ has observed labels and the required group comparisons are valid. For example, an expression-only sample contributes to the expression loss and the expression one-vs-rest fairness coordinates, but not to valence–arousal or AU losses unless those labels are observed.

Preprocessing and split protocol. Faces are aligned when landmarks are available, center-cropped, resized to 112×112, and normalized. We use the train/validation/test partitions from the demographic annotation protocol [21, 23]. To avoid cross-dataset leakage, we remove samples from the training

split of one affective dataset if they appear in the validation or test split of another. This preprocessing and split protocol is applied consistently to FairMT and all baselines.

F.1.3 Toxicity and Social Bias Analysis

We use Civil Comments from WILDS [30] for text-based heterogeneous MTL. Each comment is annotated with toxicity-related labels and identity indicators. The benchmark contains 10 utility tasks: nine binary toxicity-related detection tasks and one scalar toxicity regression task. Since each binary task contributes one fairness coordinate and the toxicity regression task contributes one regression fairness coordinate, this benchmark also contains 10 fairness coordinates.

The nine binary detection tasks are:

Rating, *Identity Attack*, *Obscene*, *Insult*, *Likes*, *Disagree*, *Funny*, *Wow*, and *Sad*.

The scalar regression task is the continuous toxicity score $y \in [0, 1]$.

Sensitive group construction. We use the fine-grained identity indicators in Civil Comments to define groupings for Sexual Orientation, Race, and Religion. Each grouping defines a separate map $s_i = \pi(a_i)$, and fairness metrics are computed separately for each grouping.

For Sexual Orientation, the $LGBTQ+$ group contains comments with at least one of the identity indicators $\{$*LGBTQ*, *homosexual_gay_or_lesbian*, *bisexual*, *transgender*, *other_sexual_orientation*$\}$. The complement within the valid subset forms the $non\text{-}LGBTQ+$ group. When downsampling is used to reduce majority-group imbalance, it is performed within the corresponding train/validation/test split using a fixed random seed, and the same filtered subset is used for all methods.

For Race, we keep comments with clean race-related identity annotations and map them into four mutually exclusive groups: $\{$*Asian*, *Black*, *White*, *Latino*$\}$. A sample is included in the Race-specific subset only when its race-related identity indicators identify exactly one of these groups. Samples with missing, ambiguous, or multiple conflicting race annotations are excluded from the Race-specific evaluation subset.

For Religion, we keep samples with at most one positive religion-related annotation and map them into $\{$*Christian*, *Muslim*, *No-Religion*$\}$. The *No-Religion* group contains comments without explicit religious identity annotations. When downsampling is used to reduce majority-group imbalance, it is performed within each split using a fixed random seed, and all methods use the same resulting subset.

Train/validation/test splits. We follow the official WILDS Civil Comments split protocol [30]. Each sensitive grouping induces a different valid subset because the identity annotations needed to define Sexual Orientation, Race, and Religion are not simultaneously available for all comments. Therefore, all methods are trained and evaluated separately for each sensitive grouping using the corresponding filtered train/validation/test subset. This ensures that every comparison uses identical samples, group definitions, and split boundaries.

Text preprocessing. Comments are tokenized with the DeBERTa-v3-base tokenizer and processed by the common text pipeline described in Appendix F.5. Each input sequence is padded or truncated to a selected maximum length, and attention masks are used to distinguish valid tokens from padding. The same tokenization and preprocessing protocol is used for FairMT and all baselines.

F.1.4 Grouping Map and Coordinate Interface

Across all benchmarks, fairness evaluation is defined by first fixing one grouping map $s_i = \pi(a_i) \in \mathcal{G}$. The map π may correspond to a single sensitive attribute, such as Gender, Race, Religion, or Sexual Orientation, or to an intersectional grouping such as Age$\times$Gender. Once s_i is fixed, the same output-side coordinate construction is used for all tasks. Binary detection tasks contribute coordinates in $\mathcal{Q}_{\mathrm{bin}}$, multi-class classification tasks contribute one-vs-rest coordinates in $\mathcal{Q}_{\mathrm{cls}}$, and scalar regression tasks contribute coordinates in $\mathcal{Q}_{\mathrm{reg}}$. Thus, changing the sensitive attribute changes the group labels used for comparison, but not the output-side fairness-coordinate interface.

F.2 Utility and Fairness Metrics

This section defines the utility and fairness metrics used in the experiments. Utility metrics are computed over utility tasks $t \in \mathcal{T}$, while fairness metrics are computed over output-side fairness

coordinates $q \in \mathcal{Q}$. This distinction is important because a multi-class classification utility task has one prediction head but contributes multiple one-vs-rest fairness coordinates. All metrics are computed on the test split using observed labels only. Unless otherwise stated, the table values are percentages.

We follow the notation used in the main paper: **EO** denotes the equal-opportunity TPR gap, and **EOD** denotes the equalized-odds-style gap that combines TPR and FPR disparities.

F.2.1 Utility Metrics

Accuracy for discrete outputs. For binary detection and multi-class classification tasks, we evaluate utility using Accuracy. Let $\mathcal{I}_t^{\text{test}}$ be the set of test samples with an observed label for utility task t:

$$\mathcal{I}_t^{\text{test}} = \{i : \ m_{i,t} = 1 \text{ in the test split}\}.$$

Let

$$N_t^{\text{test}} = |\mathcal{I}_t^{\text{test}}|.$$

For task t, the accuracy is

$$\text{Acc}_t = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} \mathbb{1}[\hat{y}_{i,t} = y_{i,t}].$$

For binary outputs, $\hat{y}_{i,t}$ is obtained by thresholding the predicted probability. For multi-class classification outputs, $\hat{y}_{i,t}$ is the class with the largest softmax probability.

F1 score for the Pareto analysis. The main benchmark tables use Accuracy for discrete outputs to keep the utility metric uniform across binary detection and multi-class classification. For the AU2–Arousal Pareto analysis, we additionally evaluate AU2 detection using F1. For a binary task t, define precision and recall as

$$\text{Prec}_t = \frac{\text{TP}_t}{\text{TP}_t + \text{FP}_t},$$

and

$$\text{Rec}_t = \frac{\text{TP}_t}{\text{TP}_t + \text{FN}_t},$$

where TP_t, FP_t, and FN_t denote true positives, false positives, and false negatives on the observed test samples of task t. The F1 score is

$$\text{F1}_t = \frac{2\,\text{Prec}_t\,\text{Rec}_t}{\text{Prec}_t + \text{Rec}_t}. \tag{69}$$

If the denominator in Eq. (69) is zero, the F1 score is set to zero for that task.

CCC for regression outputs. For scalar regression tasks, including Valence, Arousal, and toxicity regression, we evaluate utility using the Concordance Correlation Coefficient (CCC). For task t, define the prediction and target means over observed test samples as

$$\mu_{\hat{y}_t} = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} \hat{y}_{i,t}, \qquad \mu_{y_t} = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} y_{i,t}.$$

The corresponding variances are

$$\sigma_{\hat{y}_t}^2 = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} (\hat{y}_{i,t} - \mu_{\hat{y}_t})^2, \qquad \sigma_{y_t}^2 = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} (y_{i,t} - \mu_{y_t})^2.$$

The covariance is

$$\text{cov}(\hat{y}_t, y_t) = \frac{1}{N_t^{\text{test}}} \sum_{i \in \mathcal{I}_t^{\text{test}}} (\hat{y}_{i,t} - \mu_{\hat{y}_t})(y_{i,t} - \mu_{y_t}).$$

The CCC is

$$\text{CCC}_t = \frac{2\,\text{cov}(\hat{y}_t, y_t)}{\sigma_{\hat{y}_t}^2 + \sigma_{y_t}^2 + (\mu_{\hat{y}_t} - \mu_{y_t})^2}.$$

Higher CCC indicates better regression utility.

Utility aggregation. When a benchmark contains multiple tasks of the same output type, we average utility over the corresponding task set. For an evaluated utility-task set $\mathcal{T}_{\text{eval}}$, let Utility_t be the utility score for task t. The mean utility is

$$\text{MeanUtility} = \frac{1}{|\mathcal{T}_{\text{eval}}|} \sum_{t \in \mathcal{T}_{\text{eval}}} \text{Utility}_t.$$

For example, we average Accuracy over 29 attribute tasks on CelebA and AU utility over eight AU tasks on EmotioNet. On AffectNet-VA, we average CCC over Valence and Arousal when a single aggregate is needed.

Prediction thresholding. During training, discrete fairness coordinates are computed from soft prediction scores. During evaluation, binary decisions are obtained from thresholded predictions. For CelebA attribute detection and EmotioNet AU detection, we use a fixed threshold of 0.5. For Civil Comments detection tasks, the threshold is selected on the validation split to maximize macro-F1 and then fixed for final test evaluation. The same evaluation threshold for a task and sensitive grouping is used across all methods.

F.2.2 Fairness Metrics

Valid group cells. Fairness metrics are computed only over observed coordinate–group cells. For a fairness coordinate q, let $Y_{i,q}$ denote the induced ground-truth label or target, and let $\hat{Y}_{i,q}$ denote the corresponding prediction after applying the evaluation rule for that coordinate. For a discrete coordinate, define the positive and negative test cells for group g as

$$\mathcal{I}^+_{g,q} = \{i : \ s_i = g, \ m_{i,q} = 1, \ Y_{i,q} = 1\},$$

and

$$\mathcal{I}^-_{g,q} = \{i : \ s_i = g, \ m_{i,q} = 1, \ Y_{i,q} = 0\}.$$

The valid group sets for the TPR and FPR components are

$$\mathcal{G}^+_q = \{g \in \mathcal{G} : |\mathcal{I}^+_{g,q}| > 0\}, \qquad \mathcal{G}^-_q = \{g \in \mathcal{G} : |\mathcal{I}^-_{g,q}| > 0\}.$$

A TPR disparity is evaluated only when $|\mathcal{G}^+_q| \geq 2$, and an FPR disparity is evaluated only when $|\mathcal{G}^-_q| \geq 2$. Coordinates without enough valid groups are omitted from the corresponding aggregate metric.

Fairness for binary detection coordinates. For a binary fairness coordinate $q \in \mathcal{Q}_{\text{bin}}$, the group-wise true-positive rate is

$$\text{TPR}_{g,q} = \frac{1}{|\mathcal{I}^+_{g,q}|} \sum_{i \in \mathcal{I}^+_{g,q}} \mathbb{1}[\hat{Y}_{i,q} = 1], \qquad g \in \mathcal{G}^+_q. \tag{70}$$

The group-wise false-positive rate is

$$\text{FPR}_{g,q} = \frac{1}{|\mathcal{I}^-_{g,q}|} \sum_{i \in \mathcal{I}^-_{g,q}} \mathbb{1}[\hat{Y}_{i,q} = 1], \qquad g \in \mathcal{G}^-_q.$$

The TPR gap is

$$\Delta^{\text{TPR}}_q = \max_{g,g' \in \mathcal{G}^+_q} |\text{TPR}_{g,q} - \text{TPR}_{g',q}|.$$

The FPR gap is

$$\Delta^{\text{FPR}}_q = \max_{g,g' \in \mathcal{G}^-_q} |\text{FPR}_{g,q} - \text{FPR}_{g',q}|.$$

Following the notation in the main paper, Equal Opportunity is denoted by EO and is defined as

$$\text{EO}_q = \Delta^{\text{TPR}}_q.$$

Equalized Odds Difference is denoted by EOD and is defined as

$$\text{EOD}_q = \frac{1}{2} \left(\Delta^{\text{TPR}}_q + \Delta^{\text{FPR}}_q \right). \tag{71}$$

Lower EO and EOD indicate smaller group disparity.

Fairness coordinates for multi-class classification. For a multi-class classification task t with C_t classes, fairness is evaluated in a one-vs-rest manner. Each class $c \in \{1, \ldots, C_t\}$ induces a fairness coordinate $q = (t, c)$. The induced binary ground-truth label is

$$Y_{i,(t,c)} = \mathbb{1}[y_{i,t} = c].$$

The induced binary prediction is

$$\hat{Y}_{i,(t,c)} = \mathbb{1}[\hat{y}_{i,t} = c].$$

For each class-wise coordinate $q = (t, c)$, we compute $\mathrm{TPR}_{g,q}$, $\mathrm{FPR}_{g,q}$, EO_q, and EOD_q using Eqs. (70)–(71). The multi-class classification EO for task t is the class-wise average:

$$\mathrm{EO}_t^{\mathrm{multi}} = \frac{1}{C_t} \sum_{c=1}^{C_t} \mathrm{EO}_{(t,c)}.$$

The multi-class classification EOD for task t is

$$\mathrm{EOD}_t^{\mathrm{multi}} = \frac{1}{C_t} \sum_{c=1}^{C_t} \mathrm{EOD}_{(t,c)}.$$

This one-vs-rest protocol matches the fairness-coordinate construction used by FairMT for multi-class classification outputs.

Regression fairness audit metrics. For regression coordinates, FairMT optimizes the asymmetric error-parity coordinate defined in Appendix B. We evaluate regression fairness with three audit metrics: KS disparity, CSP disparity, and a binned EOD-style disparity. These metrics evaluate distributional and threshold-based group disparities for continuous outputs.

For a regression coordinate $q \in \mathcal{Q}_{\mathrm{reg}}$, define the valid regression group set as

$$\mathcal{G}_q^{\mathrm{reg}} = \{g \in \mathcal{G} : |\mathcal{I}_{g,q}^{\mathrm{reg}}| > 0\},$$

where

$$\mathcal{I}_{g,q}^{\mathrm{reg}} = \{i : \ s_i = g, \ m_{i,q} = 1\}.$$

A regression fairness metric is evaluated only when at least two valid groups are present.

KS disparity. Let $F_{q,g}$ be the empirical cumulative distribution function of predictions for regression coordinate q in group g:

$$F_{q,g}(z) = \frac{1}{|\mathcal{I}_{g,q}^{\mathrm{reg}}|} \sum_{i \in \mathcal{I}_{g,q}^{\mathrm{reg}}} \mathbb{1}[\hat{Y}_{i,q} \le z].$$

The pairwise Kolmogorov–Smirnov disparity is

$$\mathrm{KS}_q(g, g') = \sup_{z \in \mathbb{R}} \left| F_{q,g}(z) - F_{q,g'}(z) \right|.$$

The KS disparity for coordinate q is the worst pairwise group difference:

$$\mathrm{KS}_q = \max_{g,g' \in \mathcal{G}_q^{\mathrm{reg}}} \mathrm{KS}_q(g, g').$$

Lower KS indicates more similar prediction distributions across groups.

CSP disparity. Coverage Shift Probability (CSP) measures disparities in the probability of producing predictions above or below a fixed task-specific threshold. For each regression coordinate q, the threshold τ_q is fixed on the validation split and then kept identical for all methods in the corresponding evaluation. Let $\hat{Y}_q$ and s be the prediction and group label of an observed test example for coordinate q. For group g, the above-threshold probability is

$$P_q^{\mathrm{high}}(g) = \Pr(\hat{Y}_q \ge \tau_q \mid s = g).$$

The below-threshold probability is

$$P_q^{\mathrm{low}}(g) = \Pr(\hat{Y}_q < \tau_q \mid s = g).$$

The above-threshold CSP gap is

$$\mathrm{CSP}_q^{\mathrm{high}} = \max_{g,g' \in \mathcal{G}_q^{\mathrm{reg}}} \left| P_q^{\mathrm{high}}(g) - P_q^{\mathrm{high}}(g') \right|.$$

The below-threshold CSP gap is

$$\mathrm{CSP}_q^{\mathrm{low}} = \max_{g,g' \in \mathcal{G}_q^{\mathrm{reg}}} \left| P_q^{\mathrm{low}}(g) - P_q^{\mathrm{low}}(g') \right|.$$

The aggregate CSP in the main regression tables is

$$\mathrm{CSP}_q = \frac{1}{2}\left(\mathrm{CSP}_q^{\mathrm{high}} + \mathrm{CSP}_q^{\mathrm{low}}\right).$$

Binned EOD-style disparity. For continuous regression targets, we also use a binned equalized-odds-style audit metric. We partition the target range into B equal-width bins and use $B = 10$ throughout the experiments. Let b_q^{true} and b_q^{pred} denote the target and prediction bin indices for an observed test example. For bin k, group g, and regression coordinate q, define the bin-level true-positive rate as

$$\mathrm{TPR}_q^{\mathrm{bin}}(k,g) = \Pr(b_q^{\mathrm{pred}} = k \mid b_q^{\mathrm{true}} = k,\ s = g).$$

The bin-level false-positive rate is

$$\mathrm{FPR}_q^{\mathrm{bin}}(k,g) = \Pr(b_q^{\mathrm{pred}} = k \mid b_q^{\mathrm{true}} \neq k,\ s = g).$$

The bin-level TPR disparity is

$$\Delta_q^{\mathrm{bin,TPR}}(k) = \max_{g,g'} \left| \mathrm{TPR}_q^{\mathrm{bin}}(k,g) - \mathrm{TPR}_q^{\mathrm{bin}}(k,g') \right|,$$

where the maximum is taken over groups with valid denominators for bin k. The bin-level FPR disparity is

$$\Delta_q^{\mathrm{bin,FPR}}(k) = \max_{g,g'} \left| \mathrm{FPR}_q^{\mathrm{bin}}(k,g) - \mathrm{FPR}_q^{\mathrm{bin}}(k,g') \right|.$$

The bin-level equalized-odds-style gap is

$$\Delta_q^{\mathrm{bin}}(k) = \frac{1}{2}\left(\Delta_q^{\mathrm{bin,TPR}}(k) + \Delta_q^{\mathrm{bin,FPR}}(k)\right).$$

Let $\mathcal{K}_q^{\mathrm{valid}}$ be the set of bins for which the required group comparisons are valid. The final binned EOD-style disparity is

$$\mathrm{EOD}_q^{\mathrm{bin}} = \frac{1}{|\mathcal{K}_q^{\mathrm{valid}}|} \sum_{k \in \mathcal{K}_q^{\mathrm{valid}}} \Delta_q^{\mathrm{bin}}(k).$$

In regression tables, the EOD column contains $\mathrm{EOD}_q^{\mathrm{bin}}$.

Fairness aggregation over coordinates. Fairness is aggregated over fairness coordinates, not over utility heads. Let $\mathcal{Q}_{\mathrm{eval}} \subseteq \mathcal{Q}$ be the set of evaluated fairness coordinates for a given output type and sensitive grouping. For discrete coordinates, the mean EO is

$$\mathrm{MeanEO} = \frac{1}{|\mathcal{Q}_{\mathrm{eval}}|} \sum_{q \in \mathcal{Q}_{\mathrm{eval}}} \mathrm{EO}_q.$$

The mean EOD is

$$\mathrm{MeanEOD} = \frac{1}{|\mathcal{Q}_{\mathrm{eval}}|} \sum_{q \in \mathcal{Q}_{\mathrm{eval}}} \mathrm{EOD}_q.$$

Let $\mathcal{Q}_{\mathrm{reg,eval}}$ be the set of regression coordinates included in the evaluation. The mean KS is

$$\mathrm{MeanKS} = \frac{1}{|\mathcal{Q}_{\mathrm{reg,eval}}|} \sum_{q \in \mathcal{Q}_{\mathrm{reg,eval}}} \mathrm{KS}_q.$$

The mean CSP is

$$\mathrm{MeanCSP} = \frac{1}{|\mathcal{Q}_{\mathrm{reg,eval}}|} \sum_{q \in \mathcal{Q}_{\mathrm{reg,eval}}} \mathrm{CSP}_q.$$

The mean binned EOD-style disparity is

$$\mathrm{MeanEOD}^{\mathrm{bin}} = \frac{1}{|\mathcal{Q}_{\mathrm{reg,eval}}|} \sum_{q \in \mathcal{Q}_{\mathrm{reg,eval}}} \mathrm{EOD}_q^{\mathrm{bin}}.$$

For example, CelebA fairness is averaged over 29 binary attribute coordinates; AffectNet expression fairness is averaged over seven one-vs-rest expression coordinates; AffectNet-VA regression fairness is averaged over Valence and Arousal; and Civil Comments regression fairness is evaluated on the toxicity regression coordinate.

F.3 Model Architectures

All methods use a shared–private multi-task architecture. A shared encoder maps each input to a common representation, and task-specific heads map this representation to task outputs. The same backbone and task-head architecture are used for FairMT and all compared baselines. Method-specific components, such as AHFDA, the head-aware proxy, adversarial discriminators, or dependence penalties, are training-time mechanisms and are described separately in the implementation and baseline sections.

F.3.1 Shared–Private Multi-Task Architecture

For an input x_i, the shared encoder produces a representation

$$h_i = f_{\theta_s}(x_i) \in \mathbb{R}^D,$$

where D is the representation dimension and h_i is denoted r_i in Appendix D. Each utility task $t \in \mathcal{T}$ has a task-specific head $h_{\theta_t}^{(t)}$. The task prediction is

$$\hat{y}_{i,t} = h_{\theta_t}^{(t)}(h_i).$$

The output dimension of $h_{\theta_t}^{(t)}$ depends on the task type. Binary detection and scalar regression heads output one scalar value, while a multi-class classification head outputs a vector of class logits.

For scalar-output heads, including binary detection and scalar regression, the last affine layer can be written as

$$z_{i,t} = W_t r_{i,t} + b_t, \qquad W_t \in \mathbb{R}^{1 \times d_t^{\mathrm{hid}}},$$

where $r_{i,t} \in \mathbb{R}^{d_t^{\mathrm{hid}}}$ is the local head representation produced from h_i, and b_t is the bias. If the task head is linear, then $r_{i,t} = h_i$. If the head contains a bottleneck or nonlinear layer, $r_{i,t}$ is the hidden representation before the final affine layer.

For binary detection, the predicted positive probability is

$$p_{i,t} = \sigma(z_{i,t}).$$

For scalar regression, the model prediction is the scalar head output:

$$\hat{y}_{i,t} = z_{i,t}.$$

For a multi-class classification task with C_t classes, the final affine layer outputs class logits

$$z_{i,t} = W_t r_{i,t} + b_t, \qquad W_t \in \mathbb{R}^{C_t \times d_t^{\mathrm{hid}}},$$

and the predicted class probabilities are

$$p_{i,t,c} = \frac{\exp(z_{i,t,c})}{\sum_{c'=1}^{C_t} \exp(z_{i,t,c'})}, \qquad c = 1, \ldots, C_t.$$

The head-aware proxy in FairMT uses task-head geometry and output sensitivities. For a head affine in the shared representation, each head direction is the transpose of the corresponding row of W_t. When a head contains additional layers, we use the sample-specific Jacobian directions $H_{i,t,k}$ in Eq. (40), differentiating each output or logit through the full task head to the shared representation. The sensitivity-weighted Jacobian contributions are then summed across observed samples.

F.3.2 Visual Backbone

For the visual benchmarks, including Attribute Detection and Affective Analysis, we use ConvNeXt-Base [36] pretrained on ImageNet as the shared visual encoder. The original classification head is removed. The backbone produces a high-level visual feature map, which is aggregated by global average pooling and then passed through Layer Normalization to obtain the shared representation $h_i = f_{\theta_s}(x_i)$.

The visual model uses one task-specific prediction head per utility task. In Attribute Detection, the model has 29 binary attribute heads, each outputting one logit. In Affective Analysis, the model has one 7-class expression-classification head, two scalar regression heads for Valence and Arousal, and eight binary AU detection heads. Therefore, the visual architecture supports both the homogeneous 29-task CelebA setting and the heterogeneous affective setting without changing the shared backbone.

All visual task heads follow the same lightweight design: an optional bottleneck projection, GELU activation, dropout, and a final affine prediction layer. The final output dimension is determined by the task type: one logit for binary detection, seven logits for expression classification, and one scalar for each regression target. The same visual backbone and task-head design are used across FairMT and all visual baselines.

F.3.3 Text Backbone

For Toxicity and Social Bias Analysis, we use DeBERTa-v3-base [20] as the shared text encoder. The encoder, including token embeddings and all transformer blocks, is fully fine-tuned together with the task-specific heads. Each comment is tokenized with the DeBERTa-v3-base tokenizer, and the last-layer representation associated with the special classification token is used as the shared text representation $h_i = f_{\theta_s}(x_i)$. Tokenization length, padding, truncation, and attention-mask details are given in Appendix F.5.

The Civil Comments model has nine binary toxicity-related heads and one scalar toxicity regression head. Each binary head outputs one logit, and the toxicity regression head outputs one scalar value. All heads take the same shared text representation as input. This architecture supports heterogeneous text-based MTL while keeping a single shared language encoder.

F.3.4 Architectural Consistency Across Methods

We selected two established, high-capacity pretrained encoders, ConvNeXt-Base for vision and DeBERTa-v3-base for text. FairMT and all baselines use the same task-output architecture for the corresponding benchmark. Utility-oriented baselines such as MGDA and GradNorm use the same shared encoder and task heads as FairMT. Hybrid baselines also use the same prediction architecture and differ only in their optimization or fairness regularization. Fairness-oriented baselines may introduce auxiliary training-time modules, such as a sensitive-attribute discriminator for LNL, but the prediction backbone and task heads are kept identical for fair comparison.

At inference time, FairMT uses only the trained shared encoder and task-specific prediction heads:

$$x_i \mapsto f_{\theta_s}(x_i) \mapsto \{h_{\theta_t}^{(t)}(f_{\theta_s}(x_i))\}_{t\in\mathcal{T}}.$$

AHFDA, the head-aware proxy, active-coordinate construction, and the dual multiplier update are not used during inference. Therefore, FairMT does not add inference-time heads, fairness classifiers, or auxiliary modules to the deployed multi-task model.

F.4 Multi-Task Utility Loss

This section defines the utility losses used for all benchmarks and all compared methods. Utility losses are computed over utility tasks $t \in \mathcal{T}$, while fairness is computed separately over output-side coordinates $q \in \mathcal{Q}$. The utility loss definitions in this section are shared by FairMT and all baselines; differences between methods come from task weighting, fairness handling, and optimization, not from task-specific utility objectives.

F.4.1 Masked Multi-Task Utility Risk

For a mini-batch $\mathcal{B}$, each task t has an active sample set

$$\mathcal{B}_t = \{i \in \mathcal{B} : m_{i,t} = 1\},$$

with active sample count

$$N_{t,\mathcal{B}} = |\mathcal{B}_t| = \sum_{i \in \mathcal{B}} m_{i,t}.$$

The active utility-task set is

$$\mathcal{T}_{\mathcal{B}} = \{t \in \mathcal{T} : N_{t,\mathcal{B}} > 0\}.$$

A task with $N_{t,\mathcal{B}} = 0$ is inactive in the current mini-batch and contributes no utility gradient.

For binary detection and multi-class classification tasks, the masked empirical risk is the average of the per-sample loss over active samples:

$$\widehat{L}_t^{\mathcal{B}}(\theta) = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \ell_t(\hat{y}_{i,t}, y_{i,t}), \qquad t \in \mathcal{T}_{\mathcal{B}}.$$

This is the active-task form of Eq. (1) in the main paper. For regression tasks, the MSE component is sample-wise, while the CCC component is computed over the active mini-batch $\mathcal{B}_t$. The resulting active-batch regression loss is used as $\widehat{L}_t^{\mathcal{B}}(\theta)$.

When FairMT or a baseline forms a weighted multi-task utility objective with task weights α, the mini-batch utility term is

$$\widehat{L}_{\text{util}}^{\mathcal{B}}(\theta; \alpha) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t \widehat{L}_t^{\mathcal{B}}(\theta),$$

where inactive tasks have zero weight after expansion to the full task set.

F.4.2 Binary Detection Loss

For binary detection tasks, including CelebA facial attributes, EmotioNet action units, and Civil Comments binary labels, the task head outputs a logit

$$z_{i,t} \in \mathbb{R}.$$

The predicted positive probability is

$$p_{i,t} = \sigma(z_{i,t}) = \frac{1}{1 + \exp(-z_{i,t})}.$$

Given a binary target $y_{i,t} \in \{0, 1\}$, we use binary cross-entropy with logits:

$$\ell_t^{\text{bin}}(z_{i,t}, y_{i,t}) = -y_{i,t} \log \sigma(z_{i,t}) - (1 - y_{i,t}) \log(1 - \sigma(z_{i,t})).$$

Equivalently, the numerically stable logits form is

$$\ell_t^{\text{bin}}(z_{i,t}, y_{i,t}) = \text{softplus}(z_{i,t}) - y_{i,t} z_{i,t},$$

where

$$\text{softplus}(u) = \log(1 + \exp(u)).$$

The mini-batch binary utility loss is

$$\widehat{L}_t^{\mathcal{B},\text{bin}}(\theta) = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \ell_t^{\text{bin}}(z_{i,t}, y_{i,t}). \tag{72}$$

F.4.3 Multi-Class Classification Loss

For a multi-class classification task t with C_t classes, the task head outputs logits

$$z_{i,t} = (z_{i,t,1}, \ldots, z_{i,t,C_t}) \in \mathbb{R}^{C_t}.$$

The softmax probability for class c is

$$p_{i,t,c} = \frac{\exp(z_{i,t,c})}{\sum_{c'=1}^{C_t} \exp(z_{i,t,c'})}.$$

Given a class label $y_{i,t} \in \{1, \dots, C_t\}$, we use cross-entropy:

$$\ell_t^{\text{cls}}(z_{i,t}, y_{i,t}) = -\log p_{i,t,y_{i,t}}.$$

Equivalently,

$$\ell_t^{\text{cls}}(z_{i,t}, y_{i,t}) = -z_{i,t,y_{i,t}} + \log\left(\sum_{c=1}^{C_t} \exp(z_{i,t,c})\right).$$

The mini-batch classification utility loss is

$$\widehat{L}_t^{\mathcal{B},\text{cls}}(\theta) = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \ell_t^{\text{cls}}(z_{i,t}, y_{i,t}). \tag{73}$$

In Affective Analysis, the expression task has $C_t = 7$ classes.

F.4.4 Scalar Regression Loss

For scalar regression tasks, including Valence, Arousal, and toxicity regression, the task head outputs a scalar prediction

$$\hat{y}_{i,t} \in \mathbb{R}.$$

The active mini-batch target and prediction means are

$$\mu_{y_t}^{\mathcal{B}} = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} y_{i,t}, \qquad \mu_{\hat{y}_t}^{\mathcal{B}} = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \hat{y}_{i,t}.$$

The active mini-batch variances are

$$\left(\sigma_{y_t}^{\mathcal{B}}\right)^2 = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \left(y_{i,t} - \mu_{y_t}^{\mathcal{B}}\right)^2, \qquad \left(\sigma_{\hat{y}_t}^{\mathcal{B}}\right)^2 = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \left(\hat{y}_{i,t} - \mu_{\hat{y}_t}^{\mathcal{B}}\right)^2.$$

The active mini-batch covariance is

$$\text{cov}_{\mathcal{B}}(\hat{y}_t, y_t) = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} \left(\hat{y}_{i,t} - \mu_{\hat{y}_t}^{\mathcal{B}}\right)\left(y_{i,t} - \mu_{y_t}^{\mathcal{B}}\right).$$

The active mini-batch CCC is

$$\text{CCC}_t^{\mathcal{B}} = \frac{2\,\text{cov}_{\mathcal{B}}(\hat{y}_t, y_t)}{\left(\sigma_{\hat{y}_t}^{\mathcal{B}}\right)^2 + \left(\sigma_{y_t}^{\mathcal{B}}\right)^2 + \left(\mu_{\hat{y}_t}^{\mathcal{B}} - \mu_{y_t}^{\mathcal{B}}\right)^2 + \varepsilon_{\text{ccc}}},$$

where $\varepsilon_{\text{ccc}} > 0$ is a small numerical constant.

The active mini-batch MSE is

$$\widehat{L}_t^{\mathcal{B},\text{MSE}}(\theta) = \frac{1}{N_{t,\mathcal{B}}} \sum_{i \in \mathcal{B}_t} (\hat{y}_{i,t} - y_{i,t})^2.$$

The CCC loss is

$$\widehat{L}_t^{\mathcal{B},\text{CCC}}(\theta) = 1 - \text{CCC}_t^{\mathcal{B}}.$$

The regression utility loss combines MSE and CCC loss:

$$\widehat{L}_t^{\mathcal{B},\text{reg}}(\theta) = \frac{1}{2}\left(\widehat{L}_t^{\mathcal{B},\text{MSE}}(\theta) + \widehat{L}_t^{\mathcal{B},\text{CCC}}(\theta)\right). \tag{74}$$

If $N_{t,\mathcal{B}} < 2$, the CCC term is not stable. In that case, we use the MSE component for the regression update in that mini-batch:

$$\widehat{L}_t^{\mathcal{B},\text{reg}}(\theta) = \widehat{L}_t^{\mathcal{B},\text{MSE}}(\theta), \qquad N_{t,\mathcal{B}} < 2. \tag{75}$$

F.4.5 Relation Between Regression Utility and Regression Fairness

The regression utility loss in Eq. (74) is used to optimize predictive performance. It combines MSE, which controls pointwise error, and CCC, which encourages agreement in correlation, scale, and mean. The regression fairness coordinate uses an MSE-style group error. Specifically, for a regression coordinate q, the group-wise fairness error is based on a decomposable sample-wise squared error:

$$\ell_{\text{reg-fair}}(\hat{y}_{i,q}, y_{i,t(q)}) = (\hat{y}_{i,q} - y_{i,t(q)})^2. \tag{76}$$

CCC is useful as a utility metric and training signal, but it is a batch-level statistic and is not naturally decomposable into group-wise sample contributions. The MSE-style error in Eq. (76) gives a stable group-wise error statistic for the asymmetric regression fairness coordinate in Appendix B. Thus, regression utility and regression fairness are coupled through the same predictions, but they use different loss statistics for their respective roles.

F.4.6 Benchmark-Specific Utility Objectives

The utility losses above instantiate the three benchmark settings as follows. Attribute Detection uses the binary detection loss in Eq. (72) for all 29 CelebA attribute tasks. Affective Analysis uses the multi-class classification loss in Eq. (73) for expression classification, the regression loss in Eq. (74) for Valence and Arousal, and the binary detection loss in Eq. (72) for the eight AU tasks. Toxicity and Social Bias Analysis uses the binary detection loss for the nine toxicity-related binary tasks and the regression loss for the continuous toxicity score.

F.5 Training Protocol and Implementation Details

This section describes the training protocol used for FairMT and all compared methods. The model architectures and utility losses are defined in Appendices F.3 and F.4; this section focuses on implementation choices, partial-label handling, fairness-coordinate computation, normalization and tolerance calibration, inner optimization solvers, optimization schedules, and model selection. Unless otherwise stated, all methods use the same data splits, preprocessing, backbone, task heads, utility losses, optimizer, batch size, and training budget.

All models are implemented in PyTorch with CUDA acceleration. Unless otherwise stated, experiments use Python 3.10, PyTorch 2.9, cuDNN, automatic mixed precision (AMP), and a single NVIDIA A100 GPU. AMP is enabled during training to reduce memory consumption and improve throughput. FairMT and all baselines are trained in the same software and hardware environment. Wall-clock measurements in the efficiency study include forward propagation, utility-loss computation, fairness-coordinate computation, AHFDA allocation, head-aware task weighting, backward propagation, and optimizer updates.

F.5.1 Input Preprocessing

Visual inputs. For Attribute Detection and Affective Analysis, faces are aligned when landmarks are available, center-cropped, resized to 112×112, and normalized using the same statistics for all methods. During training, we use standard visual augmentations, including random crop, random horizontal flip, and color jitter when applicable. Validation and test images are processed deterministically without random augmentation. The same preprocessing pipeline is used for CelebA, AffectNet, AffectNet-VA, and EmotioNet.

Text inputs. For Toxicity and Social Bias Analysis, comments are tokenized with the DeBERTa-v3-base SentencePiece tokenizer. Each sequence is padded or truncated to a maximum length $L_{\max} \in \{128, 256\}$, selected on the validation split and then fixed for all methods under the corresponding sensitive grouping. Attention masks are used to distinguish valid tokens from padding. Longer sequences are truncated from the end. The pooled representation associated with the special classification token is used as the shared text representation for all toxicity-related heads.

F.5.2 Partial-Label Handling

For partially supervised benchmarks, especially Affective Analysis, each sample may contain labels for only a subset of tasks. We use the observation mask $m_{i,t} \in \{0, 1\}$ for both utility optimization

and fairness-coordinate estimation. For a mini-batch $\mathcal{B}$, the active sample set of task t is

$$\mathcal{B}_t = \{i \in \mathcal{B} : m_{i,t} = 1\}.$$

The active utility-task set is

$$\mathcal{T}_{\mathcal{B}} = \{t \in \mathcal{T} : |\mathcal{B}_t| > 0\}.$$

If $t \notin \mathcal{T}_{\mathcal{B}}$, task t contributes no utility loss or utility gradient in that update.

The same mask is inherited by fairness coordinates. For a coordinate q, with parent utility task $t(q)$, we use

$$m_{i,q} = m_{i,t(q)}.$$

Thus, group-wise TPR/FPR estimates, one-vs-rest class rates, and regression errors are computed only over observed labels. A missing task annotation contributes neither a utility gradient nor a fairness-gradient signal.

F.5.3 Fairness-Coordinate Computation

For each mini-batch, FairMT constructs the active fairness-coordinate set

$$\mathcal{Q}_{\mathcal{B}} = \{q \in \mathcal{Q} : q \text{ has valid group comparisons in } \mathcal{B}\}.$$

The number of active coordinates is

$$M_{\mathcal{B}} = |\mathcal{Q}_{\mathcal{B}}|.$$

Discrete coordinates require valid TPR and/or FPR group comparisons, and regression coordinates require at least two valid groups. The detailed validity rules are given in Appendix A and Appendix B.

The raw active violation vector is

$$\Psi_{\mathcal{B}}(\theta) = (\psi_q(\theta))_{q \in \mathcal{Q}_{\mathcal{B}}}.$$

For discrete coordinates, we use the active-component asymmetric violation in Eq. (3). Unless otherwise stated, we set $\beta = 1$ during training, so both TPR and FPR components are used when they are observable. This corresponds to an equalized-odds-style training coordinate. Evaluation also includes equal-opportunity gaps.

For regression coordinates, we use the asymmetric error-parity violation in Eq. (4), with MSE as the base group error after target normalization. All one-sided references, including best TPR, lowest FPR, and lowest regression error, are detached from backpropagation. The reference values affect the forward violation but do not receive direct gradients through the reference statistic.

F.5.4 Coordinate-Scale Calibration

Raw fairness coordinates can have different numerical scales across output types. To prevent raw scale from dominating the global allocation, FairMT applies AHFDA to normalized active violations. We first train a utility-only MTL calibration model θ_{MTL} using the same task losses and masks but without fairness constraints. For each coordinate q, let $\mathcal{V}_q$ be the set of validation mini-batches in which q is active. In these experiments, $\mathcal{V}_q$ is nonempty for every evaluated coordinate. The fixed coordinate scale is

$$\nu_q = \max\left\{\frac{1}{|\mathcal{V}_q|}\sum_{\mathcal{B}\in\mathcal{V}_q}\psi_{q,\mathcal{B}}(\theta_{\mathrm{MTL}}), \nu_{\min}\right\},$$

where $\nu_{\min} > 0$ is a small numerical floor. The scales ν_q are fixed after calibration and treated as constants during FairMT training. Appendix K.6 examines how alternative scale settings affect the AHFDA allocation and final utility and fairness.

For an active mini-batch, the normalized active violation vector is

$$\bar{\Psi}_{\mathcal{B}}(\theta) = \left(\bar{\psi}_q(\theta)\right)_{q\in\mathcal{Q}_{\mathcal{B}}} = \left(\frac{\psi_q(\theta)}{\nu_q}\right)_{q\in\mathcal{Q}_{\mathcal{B}}}.$$

AHFDA, the dual update, and the head-aware fairness sensitivities are computed using $\bar{\Psi}_{\mathcal{B}}(\theta)$, so the global constraint allocates pressure according to the relative sizes of normalized violations.

F.5.5 Fairness-Tolerance Calibration

The fairness tolerance ϵ is calibrated from the utility-only MTL model. Let $\mathcal{V}_+$ denote validation mini-batches with at least one active fairness coordinate. For each $\mathcal{B} \in \mathcal{V}_+$, we compute an AHFDA estimate on the normalized active vector of the calibration model. The validation reference level is

$$\widehat{\Phi}_{\mu}^{\mathrm{MTL}} = \frac{1}{|\mathcal{V}_+|} \sum_{\mathcal{B} \in \mathcal{V}_+} \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}} \left(\bar{\Psi}_{\mathcal{B}}(\theta_{\mathrm{MTL}}); \bar{w}_{\mathcal{B}} \right).$$

The global tolerance is set as

$$\epsilon = \gamma \widehat{\Phi}_{\mu}^{\mathrm{MTL}}. \tag{77}$$

For each benchmark, we restricted candidate values of γ to those within a predefined acceptable range of validation utility and selected the one with the lowest validation AHFDA. The selected values were $\gamma = 0.10$ for Attribute Detection and Affective Analysis and $\gamma = 0.50$ for Toxicity and Social Bias Analysis. We fixed these values before test evaluation. Appendix K.2 examines the training response to the two tolerance factors.

F.5.6 AHFDA Inner Solver and Dual Update

For each mini-batch with $M_{\mathcal{B}} > 0$, AHFDA solves a simplex problem over $\Delta_{M_{\mathcal{B}}}$ using the normalized active vector $\bar{\Psi}_{\mathcal{B}}(\theta)$. We use $\mu = 0.10$ for all benchmarks. The AHFDA Frank–Wolfe solver uses $L_w = 20$ iterations by default, exact line search, and early stopping when the Frank–Wolfe duality gap is below 10^{-4}.

Let $w_{\mathcal{B}}^{\mathrm{FW}}$ be the final Frank–Wolfe allocation. In the outer neural-network update, this allocation is detached:

$$\bar{w}_{\mathcal{B}} = \mathrm{sg}\left(w_{\mathcal{B}}^{\mathrm{FW}} \right).$$

The active-set AHFDA estimate used in the primal and dual updates is

$$\widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}} = \bar{w}_{\mathcal{B}}^{\top} \bar{\Psi}_{\mathcal{B}}(\theta) - \frac{\mu}{2} \|\bar{w}_{\mathcal{B}}\|_2^2 + \frac{\mu}{2M_{\mathcal{B}}}.$$

The mini-batch constraint residual is

$$r_{\Phi,\mathcal{B}} = \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}} - \epsilon. \tag{78}$$

When $M_{\mathcal{B}} = 0$, FairMT performs a utility-only update and carries the dual multiplier forward. Replays of the training streams show that at least 99.251% of steps on each benchmark have an active fairness coordinate (Appendix K.5). For $M_{\mathcal{B}} > 0$, FairMT updates the dual multiplier by projected ascent.

$$\lambda \leftarrow \left[\lambda + \rho_{\lambda} r_{\Phi,\mathcal{B}} \right]_+. \tag{79}$$

We use $\rho_{\lambda} = 0.10$ for all benchmarks.

F.5.7 Head-Aware Proxy Implementation

The task-combination coefficient α is computed using the head-aware QP in Eq. (8). The QP is solved over the active utility tasks $\mathcal{T}_{\mathcal{B}}$. For each active task t, we form a utility head-space direction a_t. When $M_{\mathcal{B}} > 0$, we also form a fairness head-space direction c_t from the active normalized AHFDA constraint. When $M_{\mathcal{B}} = 0$, we set $c_t = 0$ for all active utility tasks, so the head-aware QP reduces to utility-only task weighting.

For scalar-output heads, including binary detection and regression, let $H_{i,t} = \partial z_{i,t} / \partial r_i \in \mathbb{R}^D$ be the sample-specific head Jacobian direction. Over the observed subset $\mathcal{B}_t = \{i \in \mathcal{B} : m_{i,t} = 1\}$, the utility and fairness directions are

$$a_t = \sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{L}_t^{\mathcal{B}}}{\partial z_{i,t}} H_{i,t},$$

$$c_t = \sum_{i \in \mathcal{B}_t} \frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial z_{i,t}} H_{i,t}.$$

The sensitivities differentiate the masked mini-batch utility risk and the active normalized AHFDA estimate directly, including their prescribed loss- and group-level normalizations. We sum the

sensitivity-weighted Jacobians without applying an additional batch average. For regression, the loss-specific reduction follows the MSE–CCC batch objective and its small-batch rule in Appendix F.4. For multi-class classification heads, the directions sum over both observed samples and logit dimensions, following Eqs. (41) and (43).

Let $A_{\mathcal{B}}$ stack the active utility directions row-wise and $C_{\mathcal{B}}$ stack the active fairness directions row-wise. The active-task QP uses

$$K_{\mathcal{B}}^{\text{head}} = A_{\mathcal{B}} A_{\mathcal{B}}^{\top}, \qquad b_{\mathcal{B}} = \lambda A_{\mathcal{B}} C_{\mathcal{B}}^{\top} \mathbf{1}.$$

The QP is

$$\alpha_{\mathcal{B}} = \arg\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{1}{2} \alpha^{\top} K_{\mathcal{B}}^{\text{head}} \alpha + b_{\mathcal{B}}^{\top} \alpha. \tag{80}$$

We solve Eq. (80) with projection-free Frank–Wolfe updates. The default number of iterations is $L_{\alpha} = 15$, with exact line search and optional early stopping when the duality gap is below 10^{-4}. The QP is warm-started from the previous task-weight iterate when the active task set is unchanged; otherwise, it is initialized uniformly over the active task set. The resulting active-task weights are expanded back to the full T-dimensional task set by assigning zero weight to inactive tasks.

F.5.8 Implementation Details

Optimization Schedule. All methods are trained with AdamW. For visual benchmarks, including Attribute Detection and Affective Analysis, we use learning rate $\rho_{\text{enc}} = \rho_{\text{head}} = 1 \times 10^{-4}$ for both the shared encoder and task heads. For Civil Comments, we use $\rho_{\text{enc}} = \rho_{\text{head}} = 2 \times 10^{-5}$ for the shared DeBERTa encoder and task heads. AdamW uses $(\beta_1, \beta_2) = (0.9, 0.999)$ and weight decay 0.01. We apply global gradient clipping with threshold 5.0 to stabilize training when fairness constraints are active. We use cosine learning-rate decay with a linear warm-up over the first 5% of total training iterations, and decay the learning rate to 10^{-6} by the end of training.

Batch Size, Epochs, and Checkpoint Selection. For Attribute Detection and Affective Analysis, we use batch size 1024. For Civil Comments, we use batch size 120, due to the larger memory footprint of the DeBERTa-v3-base encoder. The main benchmark runs use a fixed training schedule of 60 epochs for visual benchmarks and 100 epochs for Civil Comments, with validation-selected checkpoints. For utility-only baselines, checkpoints are selected according to validation utility. For fairness-aware methods, including FairMT, LNL, UFaTE, MGDA-Mean, and MGDA-UFaTE, checkpoints are selected using a validation utility–fairness criterion: among checkpoints with competitive validation utility, we choose the checkpoint with smaller aggregated validation fairness violation. This selection rule is applied consistently across fairness-aware methods.

Evaluation Thresholds. During training, discrete fairness coordinates are computed from soft prediction scores. During evaluation, binary decisions are obtained from thresholded predictions. For CelebA attribute detection and EmotioNet AU detection, we use a fixed threshold of 0.5 for all methods and tasks. For Civil Comments detection tasks, the threshold is selected on the validation split to maximize macro-F1 and then fixed for final test evaluation. For regression fairness audits, the task-specific thresholds used for CSP are fixed on the validation split and shared across methods. For the binned equalized-odds-style regression audit, the continuous target range is uniformly partitioned into $B = 10$ bins.

Random Seeds and Uncertainty. We fix random seeds for dataset shuffling, model initialization, stochastic data augmentation, and any group-balancing subsampling used in Civil Comments. Unless otherwise stated, the main benchmark results use the validation-selected checkpoint for each method and sensitive grouping. Component ablations with $\pm$ values are averaged over five random seeds, and the uncertainty is the standard deviation across seeds.

Fairness Grouping and Baseline Training Protocol. For each fairness evaluation, the sensitive grouping $s_i = \pi(a_i)$ is constructed according to Appendix F.1. Utility-only MTL baselines do not use group labels during training and can be evaluated under multiple sensitive groupings after training when the underlying data subset is the same. Fairness-aware methods are trained separately for each sensitive grouping because their fairness objectives depend on the specified group map. For Civil Comments, each sensitive attribute induces a different valid subset because the identity annotations needed to define Sexual Orientation, Race, and Religion are not simultaneously available for all comments. Therefore, all methods are trained and evaluated on the corresponding valid subset for the

sensitive grouping under evaluation. This protocol ensures that comparisons are made under identical group definitions, valid sample subsets, and train/validation/test splits.

Implementation Consistency Across Baselines. MGDA, GradNorm, LNL, UFaTE, MGDA-Mean, MGDA-UFaTE, and FairMT all use the same backbone, preprocessing, task heads, utility losses, optimizer, batch size, and training schedule whenever applicable. Method-specific hyperparameters are selected on the validation split using the same model-selection protocol for all fairness-aware methods.

F.6 Baseline Methods

We compare FairMT with representative methods from three families: utility-oriented multi-task optimization, fairness-oriented representation or dependence reduction, and hybrid fair-MTL methods.

Across tables, we use † to denote utility-oriented MTL baselines, * to denote fairness-oriented baselines, and $^{\dagger *}$ to denote hybrid fair-MTL baselines. The main compared methods are MGDA, GradNorm, LNL, UFaTE, MGDA-Mean, and MGDA-UFaTE. AHFDA-MGDA is used only in the component ablation to isolate the effect of AHFDA when plugged into an MGDA-style optimizer.

F.6.1 Common Training and Grouping Protocol

The cross-benchmark baselines described in this subsection are adapted to the same heterogeneous MTL setting as FairMT. They use the same utility-task set $\mathcal{T}$, the same task masks $m_{i,t}$, and the same masked utility losses described in Appendix F.4. When a method uses fairness information, it is trained with the same sensitive grouping map $s_i = \pi(a_i)$ used by FairMT for the corresponding evaluation.

Utility-oriented baselines do not use group labels during training. For Attribute Detection and Affective Analysis, the same trained utility-only model can be evaluated under multiple sensitive groupings because the demographic annotations coexist in the same benchmark. In the valence–arousal evaluation, MGDA and GradNorm use separate checkpoints for Age, Gender, and Race. For Toxicity and Social Bias Analysis, different sensitive groupings induce different valid subsets of Civil Comments. Therefore, utility-oriented baselines are trained separately on each valid Civil Comments subset so that the training and evaluation data match the corresponding grouping protocol.

Fairness-oriented and hybrid fair-MTL baselines use group labels during training. They are trained separately for each sensitive grouping, exactly as FairMT is. For Civil Comments, this means that each method is trained and evaluated on the valid subset associated with the selected grouping. In the current benchmark construction, the Race grouping uses the four mutually exclusive groups $\{\textit{Asian}, \textit{Black}, \textit{White}, \textit{Latino}\}$. This ensures that all methods are compared under identical data splits, group definitions, and valid sample subsets.

Method-specific hyperparameters are selected on the validation split using the same utility–fairness model-selection protocol described in Appendix F.5. For fairness-aware methods, checkpoints are selected among competitive-utility checkpoints by choosing the one with smaller aggregated validation disparity.

F.6.2 Utility-Oriented MTL Baselines

Utility-oriented MTL baselines optimize task performance but do not explicitly enforce group fairness constraints. They test whether standard task-gradient coordination is sufficient to reduce group disparities.

MGDA (†)**.** The Multiple Gradient Descent Algorithm (MGDA) [13, 48] treats MTL as a multi-objective optimization problem over task losses. At each mini-batch, MGDA computes task gradients with respect to the shared parameters and solves a simplex-constrained minimum-norm problem over the active utility-task set

$$\mathcal{T}_{\mathcal{B}} = \left\{ t \in \mathcal{T} : \sum_{i \in \mathcal{B}} m_{i,t} > 0 \right\}.$$

Let

$$g^{s}_{t,\mathcal{B}} = \nabla_{\theta_s} \widehat{L}^{\mathcal{B}}_{t}(\theta)$$

be the shared-parameter utility gradient for active task t. MGDA computes

$$\alpha_{\mathcal{B}}^{\mathrm{MGDA}} = \arg\min_{\alpha \in \Delta_{T_{\mathcal{B}}}} \frac{1}{2} \left\| \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t g_{t,\mathcal{B}}^{s} \right\|_2^2 .$$

The resulting utility objective is

$$\mathcal{J}_{\mathcal{B}}^{\mathrm{MGDA}}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t^{\mathrm{MGDA}} \widehat{L}_t^{\mathcal{B}}(\theta).$$

MGDA balances task utilities but does not use sensitive attributes, fairness coordinates, AHFDA, or the head-aware fairness proxy.

GradNorm (†)**.** GradNorm [8] adaptively reweights task losses to balance gradient magnitudes across tasks. It maintains task weights ω_t^{GN} and encourages tasks to train at comparable relative rates. Let

$$G_t = \left\| \nabla_{\theta_s} \left(\omega_t^{\mathrm{GN}} \widehat{L}_t^{\mathcal{B}}(\theta) \right) \right\|_2$$

be the weighted gradient norm for task t, and let $\bar{G}$ be the mean gradient norm over active tasks. GradNorm updates ω_t^{GN} to match a target norm proportional to each task's relative training rate. In our experiments, GradNorm uses the same masked task losses and active-task convention as FairMT. Like MGDA, GradNorm is fairness-agnostic: it does not use group labels during training and does not define output-side fairness coordinates.

F.6.3 Fairness-Oriented Baselines

Fairness-oriented baselines reduce demographic dependence or group disparity through representation-level or prediction-level regularization. Both methods use the same shared-encoder architecture as FairMT.

LNL (*)**.** Learning Not to Learn (LNL) [29] reduces demographic information in the learned representation by attaching an auxiliary sensitive-attribute predictor to the shared encoder. Let d_ω be the sensitive-attribute discriminator operating on the shared representation $f_{\theta_s}(x_i)$. The discriminator loss on a mini-batch is

$$\widehat{L}_{\mathrm{sens}}^{\mathcal{B}}(\theta_s, \omega) = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \mathrm{CE}\left(d_\omega(f_{\theta_s}(x_i)), s_i\right),$$

where CE is cross-entropy over the sensitive groups. LNL is implemented with gradient reversal: the discriminator is trained to predict s_i, while the shared encoder is trained to suppress sensitive information. A saddle-style view of the objective is

$$\min_{\theta_s, \{\theta_t\}} \max_{\omega} \left[\sum_{t \in \mathcal{T}_{\mathcal{B}}} \widehat{L}_t^{\mathcal{B}}(\theta) - \zeta_{\mathrm{LNL}} \widehat{L}_{\mathrm{sens}}^{\mathcal{B}}(\theta_s, \omega) \right],$$

where ζ_{LNL} controls the adversarial strength.

UFaTE (*)**.** UFaTE [12] improves the utility–fairness trade-off by penalizing dependence between model representations or predictions and sensitive attributes. Let $r_\theta(x_i)$ denote the representation or prediction signal used by the dependence penalty, and let $\mathcal{R}_{\mathrm{dep}}^{\mathcal{B}}(r_\theta, s)$ denote the dependence-reduction term on the mini-batch. The UFaTE-style objective is

$$\min_{\theta} \left[\sum_{t \in \mathcal{T}_{\mathcal{B}}} \widehat{L}_t^{\mathcal{B}}(\theta) + \zeta_{\mathrm{UFaTE}} \mathcal{R}_{\mathrm{dep}}^{\mathcal{B}}\left(r_\theta(x), s\right) \right],$$

where ζ_{UFaTE} controls the strength of the dependence penalty.

F.6.4 Hybrid Fair-MTL Baselines

Hybrid baselines combine utility-oriented MTL optimization with fairness regularization or fairness constraints. They test whether a direct composition of MTL optimization and fairness handling is sufficient to match FairMT.

MGDA-Mean ($^{\dagger *}$)**.** MGDA-Mean augments MGDA with a mean fairness constraint. It uses the same asymmetric violation coordinates, valid-cell construction, and coordinate-wise normalization as FairMT, but replaces AHFDA's adaptive simplex allocation with uniform averaging over active normalized coordinates.

Let $\mathcal{Q}_{\mathcal{B}}$ be the active fairness-coordinate set and $M_{\mathcal{B}} = |\mathcal{Q}_{\mathcal{B}}|$. When $M_{\mathcal{B}} > 0$, MGDA-Mean forms the normalized mean violation

$$\psi_{\mathcal{B}}^{\text{mean}}(\theta) = \frac{1}{M_{\mathcal{B}}} \sum_{q \in \mathcal{Q}_{\mathcal{B}}} \bar{\psi}_q(\theta) = \frac{1}{M_{\mathcal{B}}} \sum_{q \in \mathcal{Q}_{\mathcal{B}}} \frac{\psi_q(\theta)}{\nu_q}.$$

If $M_{\mathcal{B}} = 0$, no fairness estimate is formed and the fairness update is skipped for that mini-batch. The mean-constraint residual is

$$h_{\mathcal{B}}^{\text{mean}}(\theta) = \psi_{\mathcal{B}}^{\text{mean}}(\theta) - \epsilon_{\text{mean}}.$$

The tolerance is calibrated on validation data using the same normalized active-set convention:

$$\epsilon_{\text{mean}} = \gamma \psi_{\text{val}}^{\text{mean,MTL}}, \qquad \psi_{\text{val}}^{\text{mean,MTL}} = \frac{1}{|\mathcal{V}_+|} \sum_{\mathcal{B} \in \mathcal{V}_+} \psi_{\mathcal{B}}^{\text{mean}}(\theta_{\text{MTL}}).$$

The local MGDA-Mean objective is

$$\mathcal{J}_{\mathcal{B}}^{\text{MGDA-Mean}}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t^{\text{MGDA}} \widehat{L}_t^{\mathcal{B}}(\theta) + \lambda_{\text{mean}} \left[\psi_{\mathcal{B}}^{\text{mean}}(\theta) - \epsilon_{\text{mean}} \right].$$

MGDA-Mean is a strong diagnostic baseline because it shares FairMT's coordinate construction and normalization, but removes adaptive AHFDA allocation and the head-aware proxy.

MGDA-UFaTE ($^{\dagger *}$)**.** MGDA-UFaTE combines MGDA's utility-oriented task-gradient balancing with the UFaTE dependence-reduction regularizer. At each update, MGDA computes utility task weights over active tasks, while UFaTE adds a representation- or prediction-level dependence penalty with respect to the sensitive grouping. The objective is

$$\mathcal{J}_{\mathcal{B}}^{\text{MGDA-UFaTE}}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t^{\text{MGDA}} \widehat{L}_t^{\mathcal{B}}(\theta) + \zeta_{\text{UFaTE}} \mathcal{R}_{\text{dep}}^{\mathcal{B}} \left(r_\theta(x), s \right).$$

AHFDA-MGDA. AHFDA-MGDA is used in the component ablation to isolate the effect of AHFDA when inserted into an existing MGDA-style optimizer. It uses FairMT's asymmetric fairness coordinates, coordinate-wise normalization, and AHFDA allocation, but keeps MGDA's utility task weights instead of using the head-aware proxy. Its local objective is

$$\mathcal{J}_{\mathcal{B}}^{\text{AHFDA-MGDA}}(\theta) = \sum_{t \in \mathcal{T}_{\mathcal{B}}} \alpha_t^{\text{MGDA}} \widehat{L}_t^{\mathcal{B}}(\theta) + \lambda \left[\widehat{\Phi}_{\mu,\mathcal{B}}^{\text{FW}} \left(\bar{\Psi}_{\mathcal{B}}(\theta) \right) - \epsilon \right].$$

This baseline shows whether AHFDA itself improves fairness when plugged into a standard MTL optimizer. The comparison between AHFDA-MGDA and FairMT isolates the additional benefit of head-aware task routing.

F.6.5 Summary of Baseline Differences

The baselines differ along four axes: whether they coordinate utility tasks, whether they use group labels during training, whether they construct output-side fairness coordinates, and whether they use FairMT's adaptive allocation and head-aware routing. MGDA and GradNorm coordinate utility but do not use group labels. LNL and UFaTE use group labels but act through representation-level or dependence-reduction objectives. MGDA-Mean uses FairMT's normalized asymmetric coordinates but aggregates them uniformly. MGDA-UFaTE combines utility task balancing with representation-level dependence reduction. AHFDA-MGDA uses AHFDA but not the head-aware proxy. FairMT is the only method that combines asymmetric heterogeneous fairness coordinates, adaptive AHFDA allocation, and head-aware fairness routing within a constrained primal–dual MTL procedure.

G Additional Main Results

This section provides the full benchmark results corresponding to the summary table in the main paper. We organize the results by benchmark family. The tables compare average utility and fairness metrics under each sensitive grouping. All table values are percentages. For discrete fairness, EO denotes equal opportunity and EOD denotes equalized odds difference. For regression fairness, EOD denotes the binned equalized-odds-style audit metric defined in Appendix F.2.

G.1 Attribute Detection

Table 5 compares methods on CelebA Attribute Detection. This benchmark is the most controlled setting in our evaluation: all outputs are binary facial attributes and all retained labels are fully observed. It tests whether a shared visual encoder can support many attribute heads while controlling group disparities for each attribute.

Table 5: Full results on CelebA Attribute Detection. Accuracy is higher-is-better, while EOD and EO are lower-is-better. FairMT values are means $\pm$ standard deviations over five runs.

CelebA Attribute Detection						
Method	Acc ↑	Age		Acc ↑	Gender	
		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA	87.7	11.4	10.4	87.7	22.5	22.3
GradNorm	84.8	9.4	8.4	84.8	19.4	18.3
LNL	80.8	7.8	7.5	79.7	13.6	10.6
UFaTE	76.4	6.8	5.5	75.4	12.2	10.2
MGDA-Mean	76.9	7.7	6.3	74.6	17.8	17.2
MGDA-UFaTE	78.6	6.8	4.4	77.8	12.6	9.6
FairMT	**88.4 $\pm$ 0.3**	**3.4 $\pm$ 0.4**	**2.0 $\pm$ 0.3**	**88.1 $\pm$ 0.4**	**6.4 $\pm$ 0.5**	**3.9 $\pm$ 0.2**

Aggregate results. Utility-oriented MTL baselines are strong in prediction but weak in fairness. MGDA reaches high accuracy, but its fairness gaps remain large, especially under Gender. GradNorm reduces some gaps relative to MGDA, but it also lowers utility. This shows that task-gradient balancing alone leaves substantial group disparities, even when all tasks share the same binary output type.

Fairness-oriented baselines exhibit the opposite pattern. LNL and UFaTE reduce some group gaps compared with utility-only MTL, with large utility degradation. The hybrid MGDA-Mean uses the same normalized asymmetric coordinates as FairMT but averages them uniformly and still leaves large Gender gaps. MGDA-UFaTE improves some fairness scores while retaining much lower utility than FairMT. FairMT achieves the best accuracy under both Age and Gender and the lowest EOD and EO. Under Gender, it reduces the best non-FairMT EOD and EO from $12.2/9.6$ to $6.4/3.9$, while keeping the highest accuracy.

G.2 Affective Analysis

Table 6 compares methods on Affective Analysis. This benchmark combines three output types under partial supervision: expression classification, valence–arousal regression, and AU detection. It therefore tests whether one fairness-coordinate interface can support multi-class classification, binary and continuous outputs when labels are observed for different subsets of samples.

Aggregate results. Expression classification shows the difficulty of fairness in multi-class classification. MGDA preserves high accuracy but leaves large EOD/EO gaps, especially under Age. Fairness-oriented baselines reduce some expression disparities, but at the cost of lower accuracy. Hybrid baselines improve some groupings, but their performance remains uneven. FairMT gives the strongest overall profile: it achieves the best or tied-best expression accuracy under Age and Gender, and the lowest EOD/EO under Age, Gender, and Race. Under Race, for example, FairMT keeps accuracy within 0.1 of MGDA while reducing EOD/EO from MGDA's $13.9/13.9$ to $5.6/4.8$.

AU detection strengthens the same pattern. Although AU tasks are binary like CelebA attributes, they are trained jointly with expression classification and valence–arousal regression under partial supervision. Utility-oriented methods keep reasonable accuracy but do not consistently reduce disparities. Fairness-oriented and hybrid methods reduce some gaps, but often lose AU utility. FairMT obtains the best AU accuracy across Age, Gender, and Race and also the lowest EOD/EO

Table 6: Full results on Affective Analysis. Panel (a) covers expression classification, panel (b) AU detection, and panel (c) valence–arousal regression. FairMT values are means ± standard deviations over five runs.

(a) AffectNet Expression Classification

Method	Acc ↑	Age		Acc ↑	Gender		Acc ↑	Race	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA	59.5	23.4	23.4	**59.5**	8.6	8.6	**59.5**	13.9	13.9
GradNorm	57.5	20.3	19.6	57.5	9.1	9.1	57.5	14.9	14.8
LNL	53.6	15.1	15.9	55.4	7.1	7.0	53.6	10.9	9.9
UFaTE	51.1	12.8	12.4	53.4	6.0	6.0	54.0	8.7	8.5
MGDA-Mean	51.4	19.9	18.4	56.7	7.2	6.0	55.8	8.4	7.4
MGDA-UFaTE	55.6	15.0	13.2	56.4	6.7	6.8	56.2	7.4	7.4
FairMT	**60.6 ± 0.7**	**9.9 ± 0.1**	**8.3 ± 0.5**	**59.5 ± 0.4**	**4.0 ± 0.4**	**3.9 ± 0.3**	59.4 ± 0.4	**5.6 ± 0.2**	**4.8 ± 0.3**

(b) EmotioNet AU Detection

Method	Acc ↑	Age		Acc ↑	Gender		Acc ↑	Race	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA	81.3	19.1	19.4	81.3	6.5	5.9	81.3	18.6	18.4
GradNorm	78.4	21.4	20.4	78.4	5.8	5.6	78.4	18.5	18.4
LNL	69.5	12.9	11.1	73.5	4.5	4.3	74.8	15.4	15.1
UFaTE	67.3	11.1	10.9	77.4	3.3	4.8	77.8	14.3	13.3
MGDA-Mean	62.0	16.5	15.1	74.5	3.7	3.6	72.8	13.4	13.1
MGDA-UFaTE	69.6	12.5	11.2	75.4	3.9	3.5	74.7	14.9	13.7
FairMT	**81.6 ± 0.3**	**8.8 ± 0.8**	**7.5 ± 0.4**	**81.8 ± 0.3**	**2.3 ± 0.7**	**1.8 ± 0.2**	**81.8 ± 0.4**	**8.1 ± 0.5**	**6.8 ± 0.6**

(c) AffectNet-VA Regression

Method	CCC ↑	Age			CCC ↑	Gender			CCC ↑	Race		
		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓
MGDA	**69.3**	19.1	14.0	13.0	69.3	16.5	7.4	5.5	69.2	17.1	13.3	15.7
GradNorm	67.2	17.8	14.4	13.1	67.5	17.7	7.1	6.7	67.3	18.2	13.0	14.1
LNL	60.9	16.3	12.3	11.3	63.6	14.9	6.9	5.1	64.3	12.6	8.1	9.6
UFaTE	58.5	14.3	12.9	**8.6**	63.0	13.1	6.1	5.3	63.8	13.1	7.9	10.9
MGDA-Mean	59.0	18.5	13.8	11.5	64.1	14.9	5.5	4.7	62.1	16.6	11.9	13.6
MGDA-UFaTE	61.9	17.8	13.4	12.1	65.9	13.8	6.8	5.8	64.3	10.0	7.1	8.7
FairMT	68.9 ± 0.5	**13.9 ± 0.3**	**7.1 ± 0.4**	9.6 ± 0.4	**69.6 ± 0.9**	**10.7 ± 0.5**	**3.7 ± 0.7**	**2.2 ± 0.2**	**69.7 ± 0.3**	**6.2 ± 0.8**	**3.3 ± 0.2**	**4.9 ± 0.5**

under all three groupings. The Race setting is especially informative: FairMT reduces EOD/EO to $8.1/6.8$, whereas the best non-FairMT EOD/EO are $13.4/13.1$.

The regression panel shows that FairMT can handle continuous outputs together with discrete tasks. Utility-oriented baselines generally preserve CCC but leave larger KS, CSP, or binned EOD-style disparities. Representation-level fairness methods can reduce some regression audit metrics but often substantially weaken CCC. FairMT preserves CCC close to the best utility-oriented models and improves most regression fairness audits. Under Gender and Race, FairMT obtains the best CCC and the best KS/CSP/EOD values. Under Age, it achieves the best KS and CSP and remains second-best on binned EOD, while keeping CCC only 0.4 below MGDA.

G.3 Toxicity and Social Bias Analysis

Table 7 compares methods on Civil Comments. This benchmark tests whether FairMT transfers beyond visual MTL to text-based MTL. It also differs from the visual benchmarks because identity mentions and social-bias structure can affect group disparities in the shared language representation. The task set is heterogeneous because it combines multiple binary toxicity-related outputs with one continuous toxicity regression target.

Table 7: Full results on Civil Comments. Panel (a) covers binary toxicity-related detection and panel (b) toxicity regression. FairMT values are means ± standard deviations over five runs.

(a) Civil Comments Detection

Method	Acc ↑	LGBTQ+		Acc ↑	Race		Acc ↑	Religion	
		EOD ↓	EO ↓		EOD ↓	EO ↓		EOD ↓	EO ↓
MGDA	87.1	14.3	14.1	81.6	22.9	22.2	**84.1**	28.1	27.9
GradNorm	86.2	13.1	12.8	78.7	23.2	23.3	83.1	29.0	29.3
LNL	82.6	7.6	7.6	74.1	14.2	14.7	77.6	17.7	19.7
UFaTE	80.9	6.2	6.1	65.9	14.0	13.4	73.4	14.8	14.8
MGDA-Mean	81.9	11.8	11.9	74.4	17.8	16.5	75.1	15.7	14.8
MGDA-UFaTE	82.8	9.9	9.5	75.3	16.5	16.0	76.7	13.1	13.2
FairMT	**87.7** ± 0.4	**5.5** ± 0.5	**4.5** ± 0.2	**81.9** ± 0.3	**10.1** ± 0.6	**9.9** ± 0.3	83.8 ± 0.4	**6.7** ± 0.5	**6.0** ± 0.1

(b) Civil Comments Toxicity Regression

Method	CCC ↑	LGBTQ+			CCC ↑	Race			CCC ↑	Religion		
		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓		KS ↓	CSP ↓	EOD ↓
MGDA	73.4	34.9	12.5	27.4	69.2	42.7	18.2	38.9	**72.0**	39.8	15.8	35.3
GradNorm	71.5	39.5	14.9	26.8	68.4	41.1	18.8	35.6	69.4	39.9	22.3	38.3
LNL	67.7	28.3	9.7	23.9	65.3	32.9	15.1	27.9	66.1	29.9	9.5	21.6
UFaTE	65.9	25.5	7.3	21.7	62.6	27.8	15.2	26.1	63.7	25.4	12.8	24.3
MGDA-Mean	67.6	31.1	11.2	24.3	63.7	36.7	16.2	31.9	64.4	33.9	12.3	31.3
MGDA-UFaTE	68.9	24.9	6.0	22.8	64.1	29.2	14.5	27.3	66.5	28.4	11.8	23.3
FairMT	**73.7** ± 0.1	**21.7** ± 0.7	**4.9** ± 0.3	**19.5** ± 0.6	**69.3** ± 0.4	**23.9** ± 0.5	**10.2** ± 0.4	**21.1** ± 0.7	71.1 ± 0.6	**20.7** ± 0.7	**7.2** ± 0.3	**19.9** ± 0.6

Aggregate results. For binary detection, FairMT attains the lowest EOD and EO across all three groupings and the highest accuracy under LGBTQ+ and Race. Under Religion it stays close to MGDA's accuracy while reducing EOD and EO from $28.1/27.9$ to $6.7/6.0$. The reductions are largest where the utility-oriented baselines leave the highest disparities.

For toxicity regression, FairMT attains the highest CCC under LGBTQ+ and Race and the lowest KS, CSP and binned EOD under all three groupings. Under Religion, MGDA has the highest CCC and FairMT stays close while achieving the lowest disparities. Under Race, FairMT reduces binned EOD from 38.9 to 21.1 and slightly improves CCC relative to MGDA. The agreement between the detection and regression results extends the visual-benchmark pattern to a shared text encoder.

H Ablation and Robustness Analysis

H.1 Pareto Analysis

This subsection provides a detailed analysis of the two-task Pareto study in Fig. 3. The experiment uses AU2 detection and Arousal regression as a representative heterogeneous pair: AU2 is a binary detection task evaluated by F1 and EOD, while Arousal is a scalar regression task evaluated by CCC and KS. This setting isolates the core difficulty addressed by FairMT: utility and fairness signals come from different output types, different metrics, and different task heads, but must be coordinated through one shared encoder.

Experimental configurations. The gray points correspond to utility-weight grid search without an explicit fairness constraint. These configurations vary the scalarization between AU2 and Arousal utility losses, and therefore mainly show the utility trade-off induced by task weighting. The green points add AHFDA to the same utility-weight grid, testing whether adaptive fairness-coordinate allocation changes the attainable fairness region beyond utility scalarization alone. The remaining markers correspond to representative optimization choices: uniform utility weighting, MGDA, MGDA with AHFDA, and full FairMT. All configurations use the same two-task architecture, data split, and evaluation metrics; they differ only in how utility and fairness signals are combined during training.

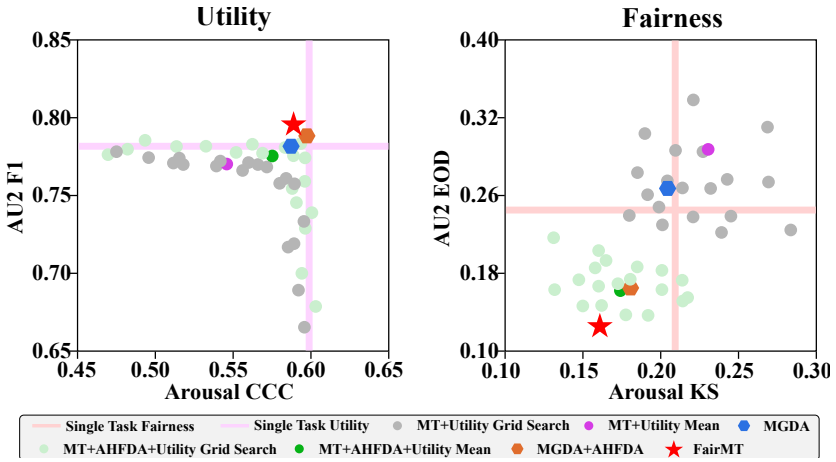

Figure 3: **Pareto analysis on AU2 detection and Arousal regression.** The pink reference lines show utility and fairness metrics from models trained independently on each task without a fairness objective. FairMT preserves strong heterogeneous utility while substantially reducing both detection and regression disparities.

Utility and fairness trade-offs. In the utility panel, the utility-only grid trades AU2 F1 against Arousal CCC, and MGDA and uniform weighting select points in that region. In the fairness panel, changing utility weights generally leaves substantial disparities. Adding AHFDA shifts the evaluated configurations toward lower AU2 EOD and Arousal KS by allocating constraint pressure across the active normalized coordinates. The comparison between MGDA and MGDA with AHFDA shows the same shift with a different utility optimizer.

Head-aware task weighting. The difference between the AHFDA-based variants and FairMT illustrates the role of the head-aware proxy. AHFDA determines which fairness coordinates receive constraint pressure, and the resulting signal passes through heterogeneous task heads before affecting the shared representation. AU2 detection and Arousal regression have different output geometries and sensitivities. FairMT uses task-head geometry to compute task-combination weights that account for these differences. It attains the highest AU2 F1 and lowest AU2 EOD among the plotted configurations. Compared with AHFDA-MGDA, it reduces Arousal KS at a small cost in CCC. The plotted trade-off shows how fairness allocation and task weighting can improve disparities while retaining strong utility across both outputs.

H.2 Component Contribution

This section isolates the contribution of FairMT's main optimization components on CelebA Attribute Detection with Gender as the sensitive grouping. We use this benchmark for component analysis because it contains 29 fully supervised binary tasks. The shared output type and complete labels let us examine task coordination, fairness allocation and head-aware routing with the supervision held fixed.

Table 8 compares utility, fairness and training behavior. Accuracy, EOD and EO are averaged over the 29 attributes. Appendix H.4 compares GFLOPs and time per epoch.

Fairness allocation. Utility-only single-task models, MGDA and vanilla MT retain substantial disparities (Rows 1, 2 and 4). Adding AHFDA to MGDA leaves accuracy at 87.7% while reducing EOD from 22.5 to 8.2 and EO from 22.3 to 6.9 (Rows 2–3). Adding AHFDA to vanilla MT shows the same pattern of disparity reduction (Rows 4 and 6). Both comparisons connect the reduction to the pressure AHFDA allocates to high-violation coordinates.

Head-aware weighting. With a utility-only objective, the proxy increases accuracy from 85.1% to 87.8% and leaves substantial disparities (Rows 4–5). With AHFDA active, the proxy adjusts the utility mixture in light of the fairness direction (Section 3.3). Adding it to MT with AHFDA raises accuracy by 2.1 points and reduces EOD from 8.5 to 6.4 (Rows 6–7). The paired variants distinguish the roles of the two components: AHFDA directs constraint pressure across coordinates, and the proxy forms a task mixture that accounts for that pressure. Full FairMT also improves all three metrics over AHFDA-MGDA.

Table 8: Component ablation on CelebA Attribute Detection with Gender as the sensitive grouping. Accuracy is higher-is-better, while EOD and EO are lower-is-better. STL denotes single-task learning. The table also includes GFLOPs and wall-clock seconds per epoch. Checkpoint epochs correspond to the checkpoints selected on the validation set. Accuracy, EOD and EO are means ± standard deviations.

Idx	Name	Method	Acc↑	EOD↓	EO↓	GFLOPs↓	Checkpoint Epoch	Training Time (s/Epoch)↓
1	STL	Utility-only STL	87.3 ± 0.5	26.7 ± 1.1	24.1 ± 0.9	13.1	45	197
2	MGDA	MGDA	87.7 ± 0.4	22.5 ± 0.8	22.3 ± 0.7	18.5	87	273
3	AHFDA-MGDA	MGDA + AHFDA	87.7 ± 0.3	8.2 ± 0.6	6.9 ± 0.5	18.7	89	280
4	MT	MT	85.1 ± 0.4	25.3 ± 0.9	22.6 ± 1.2	15.4	103	225
5	FairMT w/o AHFDA	MT + head-aware proxy	87.8 ± 0.7	24.7 ± 0.6	23.4 ± 0.2	15.8	65	247
6	FairMT w/o head-aware proxy	MT + AHFDA	86.0 ± 0.3	8.5 ± 0.8	6.3 ± 0.7	15.6	105	241
7	**FairMT (full)**	**MT + AHFDA + head-aware proxy**	**88.1 ± 0.4**	**6.4 ± 0.5**	**3.9 ± 0.2**	**16.1**	**68**	**258**

H.3 Hyperparameter Sensitivity

Table 9 varies the FairMT hyperparameters that control fairness allocation, constraint strength, and inner-solver accuracy, including the AHFDA regularization μ, the tolerance factor γ, the head-aware proxy iteration budget L_α, the AHFDA iteration budget L_w, and the dual learning rate ρ_λ. Other implementation quantities are fixed throughout the experiments. The discrete-coordinate mixing parameter is fixed to $\beta = 1$, corresponding to an equalized-odds-style training coordinate. Coordinate scales ν_q are calibrated once from the utility-only MTL model and are not tuned. The Frank–Wolfe tolerance, optimizer settings, batch sizes, thresholds, and architecture parameters are fixed as described in Appendix F.5.

This hyperparameter study uses the same coordinate-wise normalization protocol as FairMT training. Raw asymmetric violations ψ_q are divided by fixed validation-calibrated coordinate scales ν_q, and AHFDA is applied to the normalized active vector

$$\bar{\Psi}_{\mathcal{B}}(\theta) = \left(\frac{\psi_q(\theta)}{\nu_q} \right)_{q \in \mathcal{Q}_{\mathcal{B}}}.$$

The tolerance is calibrated from an unconstrained utility-only MTL model using

$$\epsilon = \gamma \widehat{\Phi}_{\mu}^{\mathrm{MTL}}, \tag{81}$$

where $\widehat{\Phi}_{\mu}^{\mathrm{MTL}}$ is the validation active-set AHFDA estimate computed on normalized violations. Unless otherwise stated, the default configuration is $\mu = 0.10$, $\gamma = 0.10$, $L_\alpha = 15$, $L_w = 20$, and $\rho_\lambda = 0.10$.

Allocation and constraint tolerance. The regularization parameter μ controls how concentrated the allocation is across active violations. A smaller μ concentrates pressure on high-violation coordinates, while a larger μ spreads that pressure more evenly. Both the most concentrated and the most diffuse settings yield higher EOD and EO than the intermediate setting $\mu = 0.10$ (Rows 1–3). The tolerance factor γ scales the budget in Eq. (81). A smaller γ imposes a stricter constraint, and a larger γ permits a larger aggregate violation. Both the tighter and the looser tested tolerances yield larger disparities than $\gamma = 0.10$, and the tighter tolerance also lowers accuracy (Rows 4–6). The sweep supports the intermediate settings for both how fairness pressure is distributed and how often it is increased.

Frank–Wolfe budgets. The head-aware QP and AHFDA allocation both improve from their smallest iteration budgets to the default budgets of $L_\alpha = 15$ and $L_w = 20$ (Rows 7–14). Increasing either budget to 30 iterations changes the balance among metrics, with no consistent improvement in utility and fairness. The default budgets capture most of the improvement from the inner solvers in this setting.

Dual learning rate. Rows 15–17 study the dual learning rate in the projected update

$$\lambda \leftarrow \left[\lambda + \rho_\lambda \left(\widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}} - \epsilon \right) \right]_+.$$

The dual learning rate controls how far the multiplier moves in response to the current residual. At $\rho_\lambda = 0.01$, the smaller updates leave EOD and EO at 10.7 and 8.5 (Row 15). At $\rho_\lambda = 0.50$, EOD

Table 9: Hyperparameter sensitivity on CelebA Attribute Detection with Gender as the sensitive grouping. We vary one hyperparameter at a time from the default FairMT configuration. Accuracy is higher-is-better, while EOD and EO are lower-is-better. Shaded rows indicate the default CelebA configuration.

Idx	Hyperparameter	Setting	Acc↑	EOD↓	EO↓
1	AHFDA regularization μ	$\mu = 0.03$	86.9	12.7	8.4
2	AHFDA regularization μ	$\mu = 0.10$ default	88.1	6.4	3.9
3	AHFDA regularization μ	$\mu = 0.30$	87.4	9.9	6.1
4	Tolerance factor γ	$\gamma = 0.01$	87.7	7.2	5.5
5	Tolerance factor γ	$\gamma = 0.10$ default	88.1	6.4	3.9
6	Tolerance factor γ	$\gamma = 0.30$	87.4	10.5	6.3
7	Head-aware proxy iterations L_α	$L_\alpha = 5$	86.3	13.7	9.2
8	Head-aware proxy iterations L_α	$L_\alpha = 10$	87.4	7.5	5.8
9	Head-aware proxy iterations L_α	$L_\alpha = 15$ default	88.1	6.4	3.9
10	Head-aware proxy iterations L_α	$L_\alpha = 30$	87.9	6.2	4.3
11	AHFDA iterations L_w	$L_w = 5$	87.4	8.6	6.8
12	AHFDA iterations L_w	$L_w = 10$	87.7	7.7	6.0
13	AHFDA iterations L_w	$L_w = 20$ default	88.1	6.4	3.9
14	AHFDA iterations L_w	$L_w = 30$	88.0	6.9	3.7
15	Dual learning rate ρ_λ	$\rho_\lambda = 0.01$	87.6	10.7	8.5
16	Dual learning rate ρ_λ	$\rho_\lambda = 0.10$ default	88.1	6.4	3.9
17	Dual learning rate ρ_λ	$\rho_\lambda = 0.50$	86.4	6.0	3.8

and EO improve slightly over the default and accuracy falls to 86.4% (Row 17). The intermediate rate $\rho_\lambda = 0.10$ retains the highest accuracy with EOD and EO close to the larger rate.

H.4 Efficiency Analysis

This subsection consolidates the efficiency analysis of FairMT. FairMT is a training-time optimization framework. After training, the deployed model consists of the same shared encoder and task-specific prediction heads as the corresponding base MTL model. We therefore measure the training-time overhead introduced by fairness-coordinate estimation, AHFDA allocation and the head-aware task-weight update.

We measure training time on one NVIDIA A100 GPU for CelebA Attribute Detection with Gender as the sensitive grouping. Table 10 compares GFLOPs and wall-clock seconds per epoch from the component study in Appendix H.2.

Table 10: Efficiency summary on CelebA Attribute Detection with Gender as the sensitive grouping. GFLOPs, checkpoint epochs, and seconds per epoch follow Table 8.

Idx	Name	Method	GFLOPs ↓	Checkpoint Epoch	Training Time (s/Epoch)↓
1	MGDA	MGDA	18.5	87	273
2	AHFDA-MGDA	MGDA + AHFDA	18.7	89	280
3	MT	Vanilla MTL	15.4	103	225
4	FairMT w/o AHFDA	MT + head-aware proxy	15.8	65	247
5	FairMT w/o head-aware proxy	MT + AHFDA	15.6	105	241
6	**FairMT (full)**	**MT + AHFDA + head-aware proxy**	16.1	68	258

Inference. FairMT predicts with the same shared encoder and task heads as the base MTL model:

$$x_i \mapsto f_{\theta_s}(x_i) \mapsto \{h_{\theta_t}^{(t)}(f_{\theta_s}(x_i))\}_{t\in\mathcal{T}}.$$

Training cost. Relative to vanilla MT, full FairMT increases computation from 15.4 to 16.1 GFLOPs and wall-clock time from 225 to 258 seconds per epoch, an increase of 14.7% (Rows 3 and 6). It takes less time per epoch than MGDA and AHFDA-MGDA (273 and 280 seconds). The training

step forms active fairness coordinates, solves AHFDA over those coordinates and solves the low-dimensional head-space QP over active utility tasks. For task weighting, MGDA instead uses per-task shared-parameter gradients (Section 3.3 and Appendix D). The full model also attains the highest accuracy and lowest EOD and EO in the component ablation (Appendix H.2).

H.5 Intersectional Grouping

This subsection evaluates whether FairMT can be used with an intersectional sensitive grouping. In the main experiments, the group map $s_i = \pi(a_i)$ is fixed to one demographic axis at a time, such as Age or Gender. Here, we retain the 29 utility tasks and 29 fairness coordinates and define the group label as the Cartesian product of Age and Gender on CelebA Attribute Detection:

$$s_i = \pi_{\mathrm{Age}\times\mathrm{Gender}}(a_i) = (\pi_{\mathrm{Age}}(a_i), \pi_{\mathrm{Gender}}(a_i)) .$$

Since CelebA Age uses $\{\textit{Young}, \textit{Old}\}$ and Gender uses $\{\textit{Male}, \textit{Female}\}$, the intersectional group set is

$$\mathcal{G}_{\mathrm{Age}\times\mathrm{Gender}} = \{(\textit{Young}, \textit{Male}), (\textit{Young}, \textit{Female}), (\textit{Old}, \textit{Male}), (\textit{Old}, \textit{Female})\}.$$

Table 11: Intersectional grouping evaluation on CelebA Attribute Detection. The sensitive group label is Age×Gender. Accuracy is higher-is-better, while EOD and EO are lower-is-better.

Idx	Method	Acc↑	EOD↓	EO↓
1	LNL	71.2	17.2	14.5
2	UFaTE	70.9	16.5	14.0
3	MGDA-Mean	70.1	18.1	16.8
4	MGDA-UFaTE	72.8	15.2	13.4
5	**FairMT**	**77.3**	**9.2**	**6.8**

Comparison under joint grouping. Of the hybrid methods, MGDA-UFaTE improves both utility and fairness over MGDA-Mean and attains the highest baseline accuracy and lowest baseline EOD and EO (Table 11). Relative to MGDA-UFaTE, FairMT improves accuracy by 4.5 points and lowers EOD and EO by 6.0 and 6.6 points. This extends the single-axis results to comparisons within the finer Age and Gender partition.

I Additional Fairness Analyses

I.1 Directional Effect of Asymmetric Violations

This subsection clarifies the directional behavior of the asymmetric fairness coordinates used by FairMT. FairMT uses a detached one-sided reference so that the explicit fairness-gradient path is directed through the currently violating group statistics. For discrete coordinates, this means increasing low-TPR groups or decreasing high-FPR groups relative to the detached reference. For regression coordinates, this means reducing high-error groups relative to the detached lowest-error group. The analysis describes the explicit fairness-gradient path for one active coordinate in the current mini-batch. We suppress batch indices on group sets, rates, and raw violations. Joint updates to the shared encoder can indirectly affect all groups, including the reference group.

TPR component for discrete coordinates. For a binary or one-vs-rest multi-class classification coordinate q, let $\widehat{\mathrm{TPR}}_{g,q}$ be the soft group-wise TPR for group g, computed over the valid positive cell. FairMT defines the detached best valid TPR reference as

$$\tau_q = \mathrm{sg}\left(\max_{g\in\mathcal{G}_q^+} \widehat{\mathrm{TPR}}_{g,q}\right),$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient. The max-based one-sided TPR violation is

$$A_q^{\mathrm{TPR}} = \max_{g\in\mathcal{G}_q^+} \left[\tau_q - \widehat{\mathrm{TPR}}_{g,q}\right]_+^2 . \tag{82}$$

Let

$$\mathcal{M}_q^{\mathrm{TPR}} = \arg\max_{g \in \mathcal{G}_q^+} \left[\tau_q - \widehat{\mathrm{TPR}}_{g,q}\right]_+^2$$

be the set of groups attaining the largest positive TPR gap. At a point with a unique maximizer $g^\star \in \mathcal{M}_q^{\mathrm{TPR}}$ and positive gap

$$\Delta_{g^\star,q}^{\mathrm{TPR}} = \tau_q - \widehat{\mathrm{TPR}}_{g^\star,q} > 0,$$

the local derivative with respect to the violating group statistic is

$$\frac{\partial A_q^{\mathrm{TPR}}}{\partial \widehat{\mathrm{TPR}}_{g^\star,q}} = -2\left(\tau_q - \widehat{\mathrm{TPR}}_{g^\star,q}\right).$$

Because τ_q is detached, the reference branch contributes no gradient to the model parameters:

$$\nabla_\theta \tau_q = 0.$$

Thus, the explicit fairness gradient does not flow through the group that defines the best-TPR reference. For a positive sample i in the violating group $g^\star$, the soft TPR is an average of predicted-positive scores, so

$$\frac{\partial A_q^{\mathrm{TPR}}}{\partial p_{i,q}} = -\frac{2\left(\tau_q - \widehat{\mathrm{TPR}}_{g^\star,q}\right)}{|\mathcal{B}_{g^\star,q}^+|}.$$

In score space, a small step along the negative gradient raises the predicted-positive scores of positive samples in the low-TPR group, moving its soft TPR toward the detached reference.

If multiple groups tie for the maximum in Eq. (82), the max operation admits a subgradient supported on the tied groups. In implementation, standard automatic differentiation selects a valid subgradient. FairMT uses the max-based coordinate to focus on the largest active violation.

FPR component for discrete coordinates. For the FPR component, FairMT uses the detached lowest valid FPR reference

$$\phi_q = \mathrm{sg}\left(\min_{g \in \mathcal{G}_q^-} \widehat{\mathrm{FPR}}_{g,q}\right).$$

The max-based one-sided FPR violation is

$$A_q^{\mathrm{FPR}} = \max_{g \in \mathcal{G}_q^-} \left[\widehat{\mathrm{FPR}}_{g,q} - \phi_q\right]_+^2.$$

Let

$$\mathcal{M}_q^{\mathrm{FPR}} = \arg\max_{g \in \mathcal{G}_q^-} \left[\widehat{\mathrm{FPR}}_{g,q} - \phi_q\right]_+^2.$$

At a point with a unique maximizer $g^\star \in \mathcal{M}_q^{\mathrm{FPR}}$ and positive gap

$$\Delta_{g^\star,q}^{\mathrm{FPR}} = \widehat{\mathrm{FPR}}_{g^\star,q} - \phi_q > 0,$$

the local derivative is

$$\frac{\partial A_q^{\mathrm{FPR}}}{\partial \widehat{\mathrm{FPR}}_{g^\star,q}} = 2\left(\widehat{\mathrm{FPR}}_{g^\star,q} - \phi_q\right),$$

while the detached reference satisfies

$$\nabla_\theta \phi_q = 0.$$

For a negative sample i in the violating group $g^\star$, the soft FPR is an average of predicted-positive scores, so

$$\frac{\partial A_q^{\mathrm{FPR}}}{\partial p_{i,q}} = \frac{2\left(\widehat{\mathrm{FPR}}_{g^\star,q} - \phi_q\right)}{|\mathcal{B}_{g^\star,q}^-|}.$$

In score space, a small step along the negative gradient lowers the predicted-positive scores of negative samples in the high-FPR group, moving its soft FPR toward the detached lowest-FPR reference.

Regression coordinates. For a regression coordinate q, let $\widehat{e}_{g,q}$ be the empirical group-wise regression error. FairMT uses the detached lowest-error valid group as the reference:

$$e_q^{\mathrm{ref}} = \mathrm{sg}\left(\min_{g \in \mathcal{G}_q^{\mathrm{reg}}} \widehat{e}_{g,q}\right).$$

The raw regression violation is

$$\psi_q^{\mathrm{reg}}(\theta) = \max_{g \in \mathcal{G}_q^{\mathrm{reg}}} \left[\widehat{e}_{g,q} - e_q^{\mathrm{ref}}\right]_+^2.$$

Let

$$\mathcal{M}_q^{\mathrm{reg}} = \arg\max_{g \in \mathcal{G}_q^{\mathrm{reg}}} \left[\widehat{e}_{g,q} - e_q^{\mathrm{ref}}\right]_+^2.$$

At a point with a unique maximizer $g^\star \in \mathcal{M}_q^{\mathrm{reg}}$ and positive gap

$$\Delta_{g^\star,q}^{\mathrm{err}} = \widehat{e}_{g^\star,q} - e_q^{\mathrm{ref}} > 0,$$

the local derivative is

$$\frac{\partial \psi_q^{\mathrm{reg}}}{\partial \widehat{e}_{g^\star,q}} = 2\left(\widehat{e}_{g^\star,q} - e_q^{\mathrm{ref}}\right),$$

and

$$\nabla_\theta e_q^{\mathrm{ref}} = 0.$$

Thus, the explicit fairness-gradient path is directed through the high-error group statistic, not through the detached lowest-error reference. With MSE-style group error, this pushes predictions for the high-error group toward their targets.

Interaction with AHFDA and coordinate normalization. The directional effects above describe individual raw coordinates. FairMT does not optimize all raw coordinates uniformly. For an active mini-batch, each raw coordinate is first normalized:

$$\bar{\psi}_q(\theta) = \frac{\psi_q(\theta)}{\nu_q}, \qquad q \in \mathcal{Q}_{\mathcal{B}}.$$

AHFDA assigns a detached allocation $\bar{w}_{q,\mathcal{B}}$ to each coordinate in the normalized active vector. With the detached outer allocation convention, the AHFDA contribution has gradient

$$\frac{\partial \widehat{\Phi}_{\mu,\mathcal{B}}^{\mathrm{FW}}}{\partial \psi_q} = \frac{\bar{w}_{q,\mathcal{B}}}{\nu_q}.$$

In the primal update, this coordinate gradient is further multiplied by the dual multiplier λ. Therefore, the explicit fairness-gradient pressure on coordinate q is scaled by

$$\lambda \frac{\bar{w}_{q,\mathcal{B}}}{\nu_q}.$$

This separates two roles: the asymmetric coordinate determines which group statistic receives local correction within coordinate q, while AHFDA determines how much global constraint pressure coordinate q receives relative to other active coordinates.

I.2 Why Error-Parity Helps Regression Fairness Audits

This subsection explains why the asymmetric error-parity coordinate used for regression training is aligned with the regression fairness audit metrics used in the experiments. We show that group-wise residual control appears in natural upper bounds for these audits under standard balance and regularity conditions. This motivates the use of an MSE-style error-parity coordinate as a differentiable training surrogate, while KS, CSP, and binned EOD-style metrics remain external audit metrics.

Residual quantities. Let $Y \in \mathbb{R}$ be a scalar regression target, $\hat{Y} \in \mathbb{R}$ be the prediction, and $S \in \mathcal{G}$ be the group label induced by the selected grouping map. Define the absolute residual

$$R = |\hat{Y} - Y|.$$

For group $g \in \mathcal{G}$, define the group-wise absolute-residual risk and squared-residual risk:

$$e_g^{(1)} = \mathbb{E}[R \mid S = g], \qquad e_g^{(2)} = \mathbb{E}[R^2 \mid S = g].$$

The regression fairness coordinate in FairMT uses an MSE-style group error because it is decomposable over samples and can be estimated over valid coordinate–group cells. By Jensen's inequality,

$$e_g^{(1)} = \mathbb{E}[R \mid S = g] \le \sqrt{\mathbb{E}[R^2 \mid S = g]} = \sqrt{e_g^{(2)}}. \tag{83}$$

Thus, controlling group-wise squared residuals also controls group-wise absolute residuals up to a square-root transformation.

Asymmetric error parity. The population analogue of FairMT's asymmetric error parity (AEP) coordinate compares each group error to the best valid group. Define

$$e_{\min}^{(2)} = \min_{g \in \mathcal{G}} e_g^{(2)},$$

and

$$\delta_2 = \max_{g \in \mathcal{G}} \left(e_g^{(2)} - e_{\min}^{(2)} \right)_+ .$$

The corresponding population AEP-style violation is

$$\psi^{\mathrm{reg}} = \delta_2^2.$$

If δ_2 is small, then every group satisfies

$$e_g^{(2)} \le e_{\min}^{(2)} + \delta_2. \tag{84}$$

Combining Eq. (84) with Eq. (83) gives

$$e_g^{(1)} \le \sqrt{e_{\min}^{(2)} + \delta_2}, \qquad \forall g \in \mathcal{G}. \tag{85}$$

This shows why regression utility and regression fairness must work together. The utility loss keeps predictive error small, while the AEP coordinate reduces group error imbalance. If the best-group error $e_{\min}^{(2)}$ is large, error parity alone is not sufficient; FairMT relies on both the utility objective and the fairness constraint.

Connection to CSP. CSP measures group differences in the probability of predictions crossing a threshold. Let τ be a fixed threshold for a regression coordinate. Define the predicted above-threshold probability

$$P_\tau^{\mathrm{high}}(g) = \Pr(\hat{Y} \ge \tau \mid S = g),$$

and the target above-threshold probability

$$T_\tau(g) = \Pr(Y \ge \tau \mid S = g).$$

For a margin $r > 0$, define the target boundary mass

$$B_{\tau,r}(g) = \Pr(|Y - \tau| \le r \mid S = g).$$

A prediction can change the threshold decision relative to the target only if the target lies within distance r of the threshold or the residual exceeds r. Therefore,

$$\left| \Pr(\hat{Y} \ge \tau \mid S = g) - \Pr(Y \ge \tau \mid S = g) \right| \le B_{\tau,r}(g) + \Pr(R > r \mid S = g). \tag{86}$$

By Markov's inequality,

$$\Pr(R > r \mid S = g) \le \frac{e_g^{(1)}}{r}. \tag{87}$$

For two groups g and g', combining Eqs. (86) and (87) yields

$$\begin{aligned} &\left| P_\tau^{\mathrm{high}}(g) - P_\tau^{\mathrm{high}}(g') \right| \\ &\le |T_\tau(g) - T_\tau(g')| + B_{\tau,r}(g) + B_{\tau,r}(g') + \frac{e_g^{(1)} + e_{g'}^{(1)}}{r}. \end{aligned}$$

Using Eq. (85), we obtain

$$\begin{aligned}&\left|P_\tau^{\text{high}}(g) - P_\tau^{\text{high}}(g')\right| \\ &\leq |T_\tau(g) - T_\tau(g')| + B_{\tau,r}(g) + B_{\tau,r}(g') + \frac{2\sqrt{e_{\min}^{(2)} + \delta_2}}{r}.\end{aligned}$$

The same bound applies to below-threshold probabilities because

$$P_\tau^{\text{low}}(g) = 1 - P_\tau^{\text{high}}(g).$$

Thus, when target threshold rates are not highly imbalanced, target mass near the threshold is limited, and residual errors are small and balanced across groups, CSP disparity is controlled.

Connection to KS disparity. Let $\mathcal{L}(\hat{Y} \mid S = g)$ denote the prediction distribution for group g, and let $\mathcal{L}(Y \mid S = g)$ denote the target distribution. By the triangle inequality for Wasserstein-1 distance,

$$W_1\left(\mathcal{L}(\hat{Y} \mid S = g), \mathcal{L}(\hat{Y} \mid S = g')\right) \leq W_1\left(\mathcal{L}(Y \mid S = g), \mathcal{L}(Y \mid S = g')\right) + e_g^{(1)} + e_{g'}^{(1)}.$$

Using Eq. (85), this implies

$$\begin{aligned}&W_1\left(\mathcal{L}(\hat{Y} \mid S = g), \mathcal{L}(\hat{Y} \mid S = g')\right) \\ &\leq W_1\left(\mathcal{L}(Y \mid S = g), \mathcal{L}(Y \mid S = g')\right) + 2\sqrt{e_{\min}^{(2)} + \delta_2}.\end{aligned} \tag{88}$$

If the group-conditional prediction distributions have densities bounded by L, a standard smoothing argument gives

$$d_{\text{KS}}\left(\mathcal{L}(\hat{Y} \mid S = g), \mathcal{L}(\hat{Y} \mid S = g')\right) \leq \sqrt{2L\, W_1\left(\mathcal{L}(\hat{Y} \mid S = g), \mathcal{L}(\hat{Y} \mid S = g')\right)}. \tag{89}$$

Combining Eqs. (88) and (89) yields

$$\begin{aligned}&d_{\text{KS}}\left(\mathcal{L}(\hat{Y} \mid S = g), \mathcal{L}(\hat{Y} \mid S = g')\right) \\ &\leq \sqrt{2L\left[W_1\left(\mathcal{L}(Y \mid S = g), \mathcal{L}(Y \mid S = g')\right) + 2\sqrt{e_{\min}^{(2)} + \delta_2}\right]}.\end{aligned}$$

Therefore, when target distributions are not highly different across groups and prediction densities are regular, reducing group-wise residual errors helps reduce KS disparity between prediction distributions.

Connection to binned EOD-style audits. The binned EOD-style audit partitions the target range into bins and computes group disparities in bin-level TPR and FPR. Let $b(Y)$ and $b(\hat{Y})$ denote the true and predicted bin indices. For bin k, let $\mathcal{I}_k$ denote the corresponding interval and let $\partial\mathcal{I}_k$ be its boundary. Define the distance from the target to the bin boundary as

$$d_k(Y) = \text{dist}(Y, \partial\mathcal{I}_k).$$

For a margin $r > 0$, if $b(Y) = k$, $d_k(Y) > r$, and $R \leq r$, then $b(\hat{Y}) = k$. Hence,

$$\begin{aligned}&\Pr(b(\hat{Y}) \neq k \mid b(Y) = k, S = g) \\ &\quad\leq \Pr(d_k(Y) \leq r \mid b(Y) = k, S = g) + \Pr(R > r \mid b(Y) = k, S = g).\end{aligned} \tag{90}$$

Assume the target-bin mass is lower bounded:

$$\Pr(b(Y) = k \mid S = g) \geq p_{\min} > 0.$$

Then

$$\mathbb{E}[R \mid b(Y) = k, S = g] \leq \frac{e_g^{(1)}}{p_{\min}}.$$

By Markov's inequality,

$$\Pr(R > r \mid b(Y) = k, S = g) \leq \frac{e_g^{(1)}}{r p_{\min}}. \tag{91}$$

Combining Eqs. (90), (91), and (85) gives

$$\Pr(b(\hat{Y}) \neq k \mid b(Y) = k, S = g)$$
$$\leq \Pr(d_k(Y) \leq r \mid b(Y) = k, S = g) + \frac{\sqrt{e^{(2)}_{\min} + \delta_2}}{r p_{\min}}.$$

Thus, the bin-level TPR error is controlled by boundary mass, bin frequency, and group residual error.

For the bin-level FPR component, a false positive into bin k occurs when $b(Y) \neq k$ but $b(\hat{Y}) = k$. A false positive requires either $d_k(Y) \leq r$ or $R > r$, so the same argument applies with a positive lower bound on the out-of-bin probability $\Pr(b(Y) \neq k \mid S = g)$. For fixed bin probabilities and boundary masses, smaller group residual errors tighten the resulting upper bound on binned EOD-style disparity.

The analysis above explains the role of the regression coordinate in FairMT. The asymmetric error-parity coordinate controls group-wise MSE imbalance, while the regression utility loss controls prediction quality. These two objectives reduce group-wise residual errors, which appear in upper bounds for threshold-based CSP, distributional KS disparity, and binned EOD-style audits. AEP is a differentiable residual-control surrogate whose reductions are expected to improve these external audits under the stated balance and regularity conditions.

J Related Work

Research on algorithmic fairness and multi-task learning has progressed along largely separate lines. Fairness methods usually define group constraints for a single prediction problem, while MTL methods coordinate multiple task losses without explicitly modeling demographic disparities. FairMT is motivated by the gap between these two views: heterogeneous MTL requires fairness to be represented over output-side coordinates, aggregated across task types, and routed through shared multi-head architectures.

Fairness learning. Algorithmic fairness has been extensively studied in the single-task setting. Classical group-fairness notions such as demographic parity, equal opportunity, and equalized odds formalize disparities between demographic groups through output-side statistics [14, 19, 41]. A large body of work incorporates these criteria into learning through constraints, reductions, covariance-style penalties, or post-processing and in-processing formulations [59, 58, 60, 2]. These approaches provide principled fairness objectives, but they are usually designed around one output space and one prediction task. As a result, they do not directly specify how to compare or aggregate fairness violations across binary detection, multi-class classification, and regression heads in a shared MTL model.

Another line of work reduces demographic dependence at the representation level. Adversarial debiasing and related representation-learning methods train feature extractors to remove sensitive information from learned features, as in adversarial representation learning and learning-not-to-learn objectives [39, 29]. Dependence-reduction methods, including UFaTE-style objectives, penalize statistical dependence between representations or predictions and sensitive attributes [12]. These methods can reduce sensitive-attribute leakage, but they are output-agnostic: they do not identify which task-specific output coordinate is currently unfair, which group is under-served, or how fairness pressure should be assigned across heterogeneous heads. This is especially limiting in MTL, where globally suppressing sensitive information can also remove task-relevant structure needed by non-violating tasks or groups.

Fairness for regression is less standardized than fairness for classification. Prior work has studied regression fairness through error parity, distributional parity, threshold-based criteria, or constrained regression formulations [4, 31, 3, 9, 10, 56]. These criteria are useful for auditing continuous outputs, but they are not directly interchangeable with rate-based classification metrics. FairMT therefore does not treat raw regression and classification metrics as the same object. Instead, it constructs task-type-specific raw violation coordinates and normalizes them before global aggregation, so that regression and discrete fairness signals can participate in one constrained MTL objective without being forced into the same raw measurement scale.

Multi-task learning. Multi-task learning improves generalization by sharing representations across related tasks [7, 46, 61]. Modern MTL often uses a shared encoder with task-specific heads, which

is also the architecture considered in FairMT. A major challenge in MTL is task interference: gradients from different tasks may have different magnitudes or conflicting directions. Multi-objective optimization methods such as MGDA and Pareto MTL formulate MTL as finding Pareto-style descent directions over task losses [13, 48, 34]. Gradient-balancing and gradient-surgery methods such as GradNorm and PCGrad adjust task weights or modify gradients to reduce conflict and improve optimization stability [8, 57].

These methods are effective for coordinating task utility, but they are fairness-agnostic. They operate over task losses and task gradients, not over group-conditioned output-side disparities. Consequently, they do not specify how fairness statistics should be computed when labels are partially observed, how heterogeneous fairness metrics should be normalized and aggregated, or how fairness pressure should be propagated through task-specific heads. This distinction is central to FairMT: utility is organized over task heads, whereas fairness is organized over output-side coordinates. A method that balances task losses can still leave large group disparities because it does not know which group-conditioned coordinate is violating fairness.

Partial supervision further complicates this setting. In heterogeneous benchmarks, different datasets or samples may annotate different tasks, so some task labels are missing in each mini-batch. Utility-oriented MTL can mask missing task losses, but fairness estimation requires stronger validity checks: a group-conditioned TPR, FPR, or regression error is only meaningful when the corresponding coordinate–group cells are observed. FairMT addresses this through active-coordinate handling: fairness coordinates contribute to the AHFDA constraint only when their required group comparisons are observable.

Fairness in multi-task learning. Fairness-aware MTL remains less developed than single-task fairness or utility-oriented MTL. Existing fair-MTL or multi-label fairness methods often attach task-wise fairness penalties, adversarial debiasing objectives, or ranking-style fairness regularizers to multi-task or multi-label models [50, 45, 62]. These approaches are valuable early steps, but they typically assume homogeneous task structures, representation-level debiasing, or per-task fairness handling. They do not provide a unified output-side coordinate system for binary, multi-class classification, and regression outputs, nor do they define an adaptive global aggregation mechanism for heterogeneous fairness violations.

Hybrid approaches can add a mean fairness penalty to an MTL optimizer or combine MGDA with representation-level dependence reduction. Uniform averaging can dilute severe violations, while representation-level dependence reduction can affect task utility without targeting the outputs responsible for the disparities. FairMT maps heterogeneous outputs to normalized asymmetric violations, uses AHFDA to allocate constraint pressure across active coordinates, and routes the resulting signal through task-head geometry. This separation supports binary detection, multi-class classification, and scalar regression under full or partial supervision.

K Additional empirical evaluations

K.1 Fairness-aware single-task references

To examine FairMT's fairness formulation without cross-task interference, we evaluated one representative task from each benchmark in a strict single-task setting. FairMT uses one utility task with the same output-coordinate fairness objective and primal–dual optimisation. For each comparison, the methods shared the target task, sensitive groups, backbone, split and evaluation metrics. ERM+Fair uses the fairness-aware classification objective of FairDG [32] and appears in the two classification comparisons.

Across the three tasks, FairMT achieves the highest utility and the lowest values of every disparity metric among the compared methods (Tables 12 and 13). The gains on both classification and regression show that the output-coordinate formulation also works with a single utility task.

Table 12: Fairness-aware single-task classification on CelebA (Age $\times$ Gender) and AffectNet (Age). The AffectNet task classifies seven expressions.

	CelebA: High Cheekbones			**AffectNet: expression**		
Method	Acc ↑	EOD ↓	EO ↓	Acc ↑	EOD ↓	EO ↓
LNL	84.1	11.0	8.7	58.2	13.9	8.7
UFaTE	81.3	6.4	5.9	57.7	14.3	7.6
ERM+Fair (FairDG)	82.5	7.2	4.4	58.3	13.0	8.2
FairMT	87.2	5.3	3.2	61.8	9.3	6.1

Table 13: Fairness-aware single-task toxicity regression on Civil Comments with Race grouping.

Method	CCC ↑	KS ↓	CSP ↓	EOD ↓
LNL	65.5	31.5	15.4	27.1
UFaTE	63.6	27.1	15.9	26.8
FairMT	69.6	22.4	10.9	20.7

K.2 Cross-benchmark constraint dynamics

We compared $\gamma = 0.10$ and 0.50 while keeping the other settings fixed within each benchmark. Equation (77) sets the tolerance by scaling the utility-only validation AHFDA reference, an aggregate of normalized squared gaps. At a given batch AHFDA estimate, a smaller γ lowers the tolerance and increases the residual in Eq. (78). A positive residual increases the dual multiplier in Eq. (79), so the proportion of positive-residual batches captures the frequency of upward updates. Civil Comments used batches of 120 examples, compared with 1,024 in the visual benchmarks. Its retained group proportions include 73.0%/27.0% for LGBTQ and 43.2%/40.7%/16.1% for Religion. Batch size and group representation shape the sample support for the group comparisons within each batch.

Increasing γ from 0.10 to 0.50 reduces the positive-residual rate from 76% to 18% on CelebA and from 73% to 21% on Affective Analysis (Table 14). Civil Comments starts higher at 94% and has positive residuals on 63% of batches under the larger tolerance. Under the larger tolerance, upward multiplier updates remain more frequent on Civil Comments than on either visual benchmark.

Table 14: Positive-residual batches in matched tolerance comparisons. A positive residual satisfies $\widehat{\Phi}_{\mu,B} > \epsilon$.

Benchmark	γ	Positive-residual batches
CelebA	0.10	76%
CelebA	0.50	18%
Affective Analysis	0.10	73%
Affective Analysis	0.50	21%
Civil Comments	0.10	94%
Civil Comments	0.50	63%

K.3 Components under heterogeneous supervision

We repeated the component ablation across all tasks of Affective Analysis (Age) and Civil Comments (Race). Affective Analysis combines expression classification, AU detection and valence–arousal regression under partial labels; Civil Comments combines detection and regression. We compared utility-only MT, the variants without AHFDA or without the head-aware proxy, and full FairMT using the same predictive backbone, task heads, splits, label masks, sensitive-group definitions, training budget and validation-based model selection. Tables 15 and 16 compare the classification and regression outputs for these four variants.

Removing AHFDA largely preserves utility but leaves high disparities across output types. On Affective Analysis, expression Accuracy changes from 60.6% to 60.1%, while EOD rises from 9.9 to 22.1. Civil Comments regression shows the same separation: CCC changes from 69.3 to 69.1, while

KS rises from 23.9 to 43.1. In these heterogeneous tasks, utility coordination alone leaves group disparities that AHFDA's shared fairness signal addresses.

Removing the head-aware proxy retains much of AHFDA's fairness improvement but lowers utility in every output family. On Affective Analysis, expression Accuracy falls from 60.6% to 58.7%, and valence–arousal CCC falls from 68.9 to 67.5; on Civil Comments, regression CCC falls from 69.3 to 67.9. Full FairMT combines higher utility with lower disparities than either single-component variant in both benchmarks. These results support head-aware routing as a complement to AHFDA: coordinate-level fairness correction benefits from task coordination when its gradients pass through heterogeneous heads.

Table 15: Complete-benchmark component ablation on Affective Analysis (Age).

	Expression			**Action units**			**Valence–arousal**			
Variant	Acc ↑	EOD ↓	EO ↓	Acc ↑	EOD ↓	EO ↓	CCC ↑	KS ↓	CSP ↓	EOD ↓
Utility-only MT	58.4	22.8	21.6	79.6	19.8	18.9	67.6	18.4	13.3	13.1
Without AHFDA	60.1	22.1	21.0	81.2	19.1	18.3	68.8	19.7	14.8	14.2
Without head proxy	58.7	11.8	10.1	79.8	10.7	9.2	67.5	15.3	8.6	10.9
FairMT	60.6	9.9	8.3	81.6	8.8	7.5	68.9	13.9	7.1	9.6

Table 16: Complete-benchmark component ablation on Civil Comments (Race).

	Detection			**Regression**			
Variant	Acc ↑	EOD ↓	EO ↓	CCC ↑	KS ↓	CSP ↓	EOD ↓
Utility-only MT	79.7	24.2	23.5	68.0	41.8	18.6	37.7
Without AHFDA	81.4	23.6	22.9	69.1	43.1	19.4	39.2
Without head proxy	80.0	12.3	11.4	67.9	27.0	12.0	24.0
FairMT	81.9	10.1	9.9	69.3	23.9	10.2	21.1

K.4 Encoder transfer

We evaluated ConvNeXt-Base and DINOv2 ViT-B/14 [43] on Affective Analysis with Age grouping, and DeBERTa-v3-base and RoBERTa-base [35] on Civil Comments with Race grouping. Within each benchmark, we varied the encoder together with its native preprocessing and matched FairMT and MGDA on the remaining settings.

FairMT lowers every disparity measure relative to the matched MGDA reference across all four encoders, with utility differences from -0.4 to $+1.1$ points (Tables 17 and 18). On the alternative backbones, expression EOD with DINOv2 falls from 20.9 to 8.9, and detection EOD with RoBERTa falls from 24.7 to 12.8. The backbone affects fine-tuning cost. Across the encoders, the AHFDA and head-aware problems are sized by the active fairness coordinates and the active utility tasks, respectively (Appendix F.3.4).

Table 17: Encoder comparisons on Affective Analysis with Age grouping.

	Expression			**Action units**			**Valence–arousal**			
Encoder / method	Acc ↑	EOD ↓	EO ↓	Acc ↑	EOD ↓	EO ↓	CCC ↑	KS ↓	CSP ↓	EOD ↓
ConvNeXt-Base / MGDA	59.5	23.4	23.4	81.3	19.1	19.4	69.3	19.1	14.0	13.0
ConvNeXt-Base / FairMT	60.6	9.9	8.3	81.6	8.8	7.5	68.9	13.9	7.1	9.6
DINOv2 ViT-B/14 / MGDA	61.0	20.9	20.3	81.5	17.8	18.1	70.3	17.6	12.6	12.1
DINOv2 ViT-B/14 / FairMT	61.5	8.9	7.7	81.2	9.3	8.2	70.0	13.5	7.5	9.4

Table 18: Encoder comparisons on Civil Comments with Race grouping.

	Detection			Regression			
Encoder / method	Acc ↑	EOD ↓	EO ↓	CCC ↑	KS ↓	CSP ↓	EOD ↓
RoBERTa-base / MGDA	76.3	24.7	24.0	68.4	41.6	18.5	37.4
RoBERTa-base / FairMT	76.9	12.8	11.0	68.7	21.5	10.9	21.3
DeBERTa-v3-base / MGDA	81.6	22.9	22.2	69.2	42.7	18.2	38.9
DeBERTa-v3-base / FairMT	81.9	10.1	9.9	69.3	23.9	10.2	21.1

K.5 Active-coordinate coverage

We replayed the original training batch streams and label masks to measure the availability of fairness comparisons (Table 19). Across all optimizer steps, complete skips occur in at most 0.749% of batches. Among batches with at least one active coordinate, the mean active count is close to the full coordinate count on every benchmark. Coverage can differ across components of an active coordinate. For coordinates that include both rates, a valid FPR comparison keeps the coordinate active when its TPR component is unavailable. For rare Civil Comments attributes, the least-covered TPR component has approximately 2.7k–4.3k valid batches per original 100-epoch training run.

Table 19: Availability in replayed training streams. The active-count column contains the conditional mean followed by the total number of coordinates. TPR counts are valid batches per original training run.

Benchmark	$P(M_B = 0)$	Active count / total	Minimum TPR batches
CelebA	0.000%	$29.0/29$	8.4k
Affective Analysis	0.310%	$17.0/17$	8.9k
Civil Comments	0.749%	$\geq 9.999/10$	2.7k–4.3k

K.6 Coordinate-scale calibration

On Affective Analysis with Age grouping, we compared full-validation calibration with scales estimated from two disjoint validation halves, perturbed scales $\nu'_q = c_q \nu_q$, and unit scales ($\nu_q = 1$). Moderate perturbations used $c_q \sim \mathcal{U}(0.8, 1.2)$. Severe distortions used $\log c_q \sim \mathcal{U}(\log 0.25, \log 4)$. We recalibrated ϵ for each setting using Eq. (77), holding γ and the other training and model-selection settings fixed.

We ranked coordinates by their mean AHFDA weights over the same validation probe batches at a frozen utility-only checkpoint. Spearman's ρ measures agreement between each setting's ranking and the default ranking. The two validation halves yield correlations of 0.93–0.94, and their final models differ from the default by at most 0.5 points in utility and 1.2 points in EOD (Table 20). Moderate perturbation retains high rank agreement (0.91) and raises EOD across all three output families, including an increase from 8.8 to 10.9 for AU detection. With severe distortion or unit scales, rank agreement drops to 0.58 or 0.46, all three utility scores fall, and all three EOD scores rise. The moderate-perturbation result shows that similar rankings can coexist with worse final disparities.

Table 20: Coordinate-scale sensitivity on Affective Analysis (Age). Higher ρ indicates closer rank agreement with the default allocation.

	Allocation and utility				Fairness (EOD ↓)		
Setting	ρ	Expr. Acc	AU Acc	VA CCC	Expression	AU	VA
Default	1.00	60.6	81.6	68.9	9.9	8.8	9.6
Validation half A	0.94	60.1	81.5	69.4	9.3	7.6	8.5
Validation half B	0.93	60.8	81.6	69.3	9.7	9.2	9.2
Moderate perturbation	0.91	59.7	80.4	69.2	11.5	10.9	10.7
Severe distortion	0.58	57.3	80.3	66.6	14.4	12.6	11.4
No normalisation	0.46	55.8	79.9	65.8	15.1	13.0	12.3

K.7 Direct fair-MTL and group-reweighting comparisons

We evaluated L2T-FMT [45], FairGrad-MT and GroupDRO-MT on CelebA with all 29 fully supervised binary tasks and Gender grouping. L2T-FMT retains its teacher for selecting task-wise utility or fairness objectives. Our task-wise adaptations, FairGrad-MT and GroupDRO-MT, extend FairGrad [40] and GroupDRO [47]. Each maintains separate group weights per task and uniformly averages the 29 task objectives. FairGrad-MT reweights sensitive groups according to fairness violations, while GroupDRO-MT uses label–Gender groups and targets worst-group prediction loss. All methods shared the target attributes, groups, splits, predictive backbone, heads, training budget, metrics and validation-only model selection. The updates described above distinguish the methods in Table 3 (Section 4.2).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's main contributions. The claims are supported by the method sections, theoretical derivations in the appendix, and experiments.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The *Ethics and limitations* paragraph explains the scope of the algorithmic fairness evaluation and discusses the oversight needed for deployment.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical claims are stated, and the main proofs and derivations are provided in the appendix with numbered and cross-referenced equations.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The appendix provides dataset construction, metrics, model architectures, utility losses, training protocol, baseline definitions, hyperparameters, and model-selection rules needed to reproduce the main experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper provides detailed experimental and implementation information for reproducibility. The code is currently private.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The appendix provides dataset construction, splits, metrics, architectures, losses, optimizer settings, hyperparameters, thresholding rules, checkpoint selection, and baseline protocols.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports mean and standard deviation over repeated runs for key results and component ablations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix F.5 describes the software environment, single NVIDIA A100 GPU, mixed-precision training, and main benchmark training schedules. Appendix H.4 reports GFLOPs and per-epoch training time for CelebA with Gender grouping.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses benchmark datasets and standard model training procedures, and the paper discusses fairness, privacy, and deployment considerations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The paper reports reductions in measured group disparities. The *Ethics and limitations* paragraph discusses deployment impacts, human oversight, monitoring, and safeguards for group annotations.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the original datasets and pretrained models used in the experiments, and the experiments follow their original access terms and usage conditions.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not conduct new human-subject experiments or collect new participant data.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.