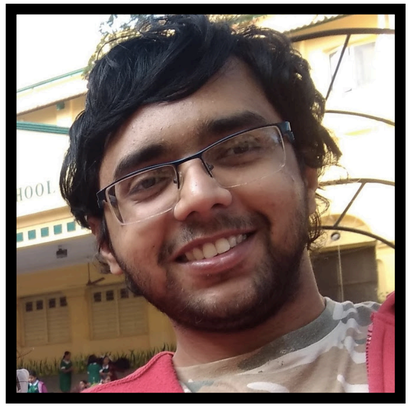
FairALM

Augmented Lagrangian Method for Training Fair Models



Vishnu Lokhande

Aditya Akash

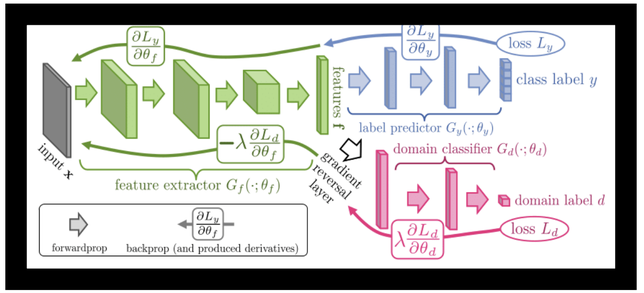
Sathya Ravi

Vikas Singh

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

General Strategies

Adversarial Methods



FairALM

Simpler Alternative

$$\min_{h \in \mathcal{H}} \quad e_h$$

$$\mu_h^{s_0} = \mu_h^{s_1}$$

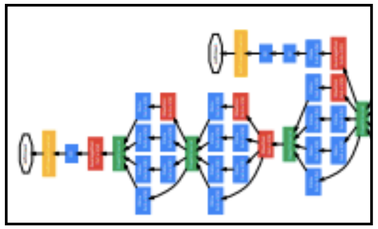
Ganin, Yaroslav, et al. "Domain-adversarial training of neural networks." *The Journal of Machine Learning Research* 17.1 (2016): 2096-2030.

- **Notations**
- The Lagrangian Formulation
- FairALM
    - Linear Classifiers
    - Deep Networks
- Experimental Results

The Expected Loss function

(x, y) : (Features, Label) → ( , 1)

s : Sensitive Attribute → Female

h : Classifier → 

$$\mathop{\mathbb{E}}_{x,y,s} \mathscr{L}(h; (x, y))$$

Fairness as Equality of Conditional Means

$$\mu_h^{s_0} = \mu_h^{s_1}$$

Demographic Parity	$\mu_h^s := \quad e_h \mid (s)$
Equality of Opportunity	$\mu_h^s := \quad e_h \mid (s, y)$
Predictive Parity	$\mu_h^s := \quad e_h \mid (s, \hat{y})$

The Constrained Optimization Problem

$$\min_{h \in \mathcal{H}} \quad e_h$$

$$\mu_h^{s_0} = \mu_h^{s_1}$$

- **Notations**
- **The Lagrangian Formulation**
- FairALM
    - Linear Classifiers
    - Deep Networks
- Experimental Results

The Lagrangian Optimization Problem

$$L(h, \lambda) = e_h + \lambda(\mu_h^{s_0} - \mu_h^{s_1})$$

$$\max_{\lambda \in R} \quad \underbrace{\min_{h \in \mathcal{H}} L(h, \lambda)}$$

Non-Smooth

Using Dual Proximal Functions

$$L_{\lambda_T}(h, \lambda) = e_h + \lambda(\mu_h^{s_0} - \mu_h^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

Using Dual Proximal Functions

$$L_{\lambda_T}(h, \lambda) = e_h + \lambda(\mu_h^{s_0} - \mu_h^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

- Notations
- The Lagrangian Formulation
- FairALM
  - Linear Classifiers
  - Deep Networks
- Experimental Results

FairALM: Linear Ensemble Classifiers

$$\mathscr{H} = \{h_1, h_2, h_3, \ldots, h_N\}$$
$$\downarrow \quad \downarrow \quad \downarrow \qquad \downarrow$$
$$e_{h_1}, e_{h_2}, e_{h_3}, \ldots, e_{h_N}$$

Agarwal, Alekh, et al. "A reductions approach to fair classification." arXiv preprint arXiv:1803.02453 (2018).

FairALM: Linear Ensemble Classifiers

$$L_{\lambda_T}(h, \lambda) = e_h + \lambda(\mu_h^{s_0} - \mu_h^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

$$q \in \Delta^N$$

$$L_{\lambda_T}(q, \lambda) = \sum_i q_i \, e_{h_i} + \sum_i q_i \, \lambda(\mu_{h_i}^{s_0} - \mu_{h_i}^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

FairALM: Linear Ensemble Classifier

$$L_{\lambda_T}(q,\lambda) = \sum_i q_i\, e_{h_i} + \sum_i q_i\, \lambda(\mu_{h_i}^{s_0} - \mu_{h_i}^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

$$\max_{\lambda \in R} \min_{q \in \Delta^N} L_{\lambda_T}(q,\lambda)$$

Linear Program in q	$\text{argmin}_i \quad L_{\lambda_t}(q_i, \lambda_t)$
Maximize the Cumulative Reward	$\lambda_{t+1} \leftarrow \lambda_t + \frac{\eta}{t}(\mu_{h_t}^{s_0} - \mu_{h_t}^{s_1})$

FairALM: Linear Ensemble Classifier

$$L_{\lambda_T}(q,\lambda) = \sum_i q_i\, e_{h_i} + \sum_i q_i\, \lambda(\mu_{h_i}^{s_0} - \mu_{h_i}^{s_1}) - \frac{1}{2\eta}(\lambda - \lambda_T)^2$$

$$\max_{\lambda \in R} \min_{q \in \Delta^N} L_{\lambda_T}(q,\lambda)$$

Linear Program in q	$\text{argmin}_i \quad L_{\lambda_t}(q_i, \lambda_t)$
Maximize the Cumulative Reward	$\lambda_{t+1} \leftarrow \lambda_t + \frac{\eta}{t}(\mu_{h_t}^{s_0} - \mu_{h_t}^{s_1})$

Convergence Analysis

After T update steps $\Longrightarrow$ ν – approximate saddle point

$$L_{\lambda_T}\left(\frac{1}{T}\sum q_t, \quad \frac{1}{T}\sum \lambda_t\right) \leq \min_q L_{\lambda_T}\left(q, \quad \frac{1}{T}\sum \lambda_t\right) + \nu$$

$$L_{\lambda_T}\left(\frac{1}{T}\sum q_t, \quad \frac{1}{T}\sum \lambda_t\right) \geq \max_\lambda L_{\lambda_T}\left(\frac{1}{T}\sum q_t\ , \lambda\right) - \nu$$

$$\nu = \mathcal{O}\left(\frac{\log^2 T}{T}\right)$$

vs. $\mathcal{O}\left(\frac{\sqrt{T}}{T}\right)$ [Agarwal et al.]

- Notations
- The Lagrangian Formulation
- FairALM
    - Linear Classifiers
    - Deep Networks
- Experimental Results

Adjustments for Deep Networks - 1

- Replace non-differentiable indicator function to a smooth surrogate function like logistic function.

- Replace Error and Conditional Means with Empirical Estimates

$$e_h \rightarrow \hat{e}_{h_w}$$

$$\mu_h \rightarrow \hat{\mu}^s_{h_w}$$

Adjustments for Deep Networks - 2

$$\max_{\lambda} \left| \min_{w} \left(\hat{e}_{h_w} + \lambda(\hat{\mu}^{s_0}_{h_w} - \hat{\mu}^{s_1}_{h_w}) - \frac{1}{2\eta}(\lambda - \lambda_t)^2 \right) \right|$$

$$\leq \left| \min_{w} \left| \underbrace{\max_{\lambda} \left(\hat{e}_{h_w} + \lambda(\hat{\mu}^{s_0}_{h_w} - \hat{\mu}^{s_1}_{h_w}) - \frac{1}{2\eta}(\lambda - \lambda_t)^2 \right)}_{\lambda \leftarrow \lambda_t + \eta(\hat{\mu}^{s0}_{h_w} - \hat{\mu}^{s1}_{h_w})} \right| \right|$$

Dual Update

$$\leq \left| \min_{w} \left(\underbrace{\hat{e}_{h_w} + (\lambda_t + \eta)\hat{\mu}^{s_0}_{h_w} - (\lambda_t - \eta)\hat{\mu}^{s_1}_{h_w}}_{\text{FairALM Objective}} \right) \right|$$

FairALM: DeepNet Classifier

1: $\lambda_0 = 0, \quad w_0 = 0$

2: **FOR** $t = 0,1,2,\ldots,T$ **DO**

3: **Sample** $z \sim$ **Data**

4: **Pick** $v_t \in \partial\left(\hat{e}_{h_w} + (\lambda_t + \eta)\hat{\mu}^{s_0}_{h_w} - (\lambda_t - \eta)\hat{\mu}^{s_1}_{h_w}\right)$

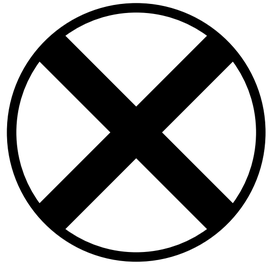
5: $w_t \leftarrow w_{t-1} - \tau v_t$ Primal Update

6: $\lambda \leftarrow \lambda_t + \eta(\hat{\mu}^{s_0}_{h_w} - \hat{\mu}^{s_1}_{h_w})$ Dual Update

7: **END**

Unconstrained

1: $w_0 = 0$
2: **FOR** $t = 0,1,2,\ldots,T$ **DO**
3: **Sample** $z \sim$ **Data**
4: **Pick** $v_t \in \partial\left(\hat{e}_{h_w}\right)$
5: $w_t \leftarrow w_{t-1} - \tau v_t$
6:
6: **END**

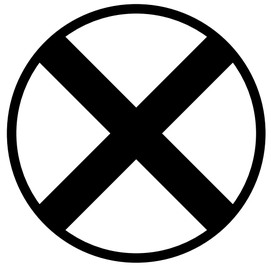


FairALM

1: $\lambda_0 = 0, \quad w_0 = 0$
2: **FOR** $t = 0,1,2,\ldots,T$ **DO**
3: **Sample** $z \sim$ **Data**
4: **Pick** $v_t \in \partial\left(\hat{e}_{h_w} + (\lambda_t + \eta)\hat{\mu}^{s_0}_{h_w} - (\lambda_t - \eta)\hat{\mu}^{s_1}_{h_w}\right)$
5: $w_t \leftarrow w_{t-1} - \tau v_t$
6: $\lambda \leftarrow \lambda_t + \eta(\hat{\mu}^{s_0}_{h_w} - \hat{\mu}^{s_1}_{h_w})$
7: **END**

Unconstrained

1: $w_0 = 0$
2: **FOR** $t = 0,1,2,\ldots,T$ **DO**
3: **Sample** $z \sim$ **Data**
4: **Pick** $v_t \in \partial\left(\hat{e}_{h_w}\right)$
5: $w_t \leftarrow w_{t-1} - \tau v_t$
6:
6: **END**



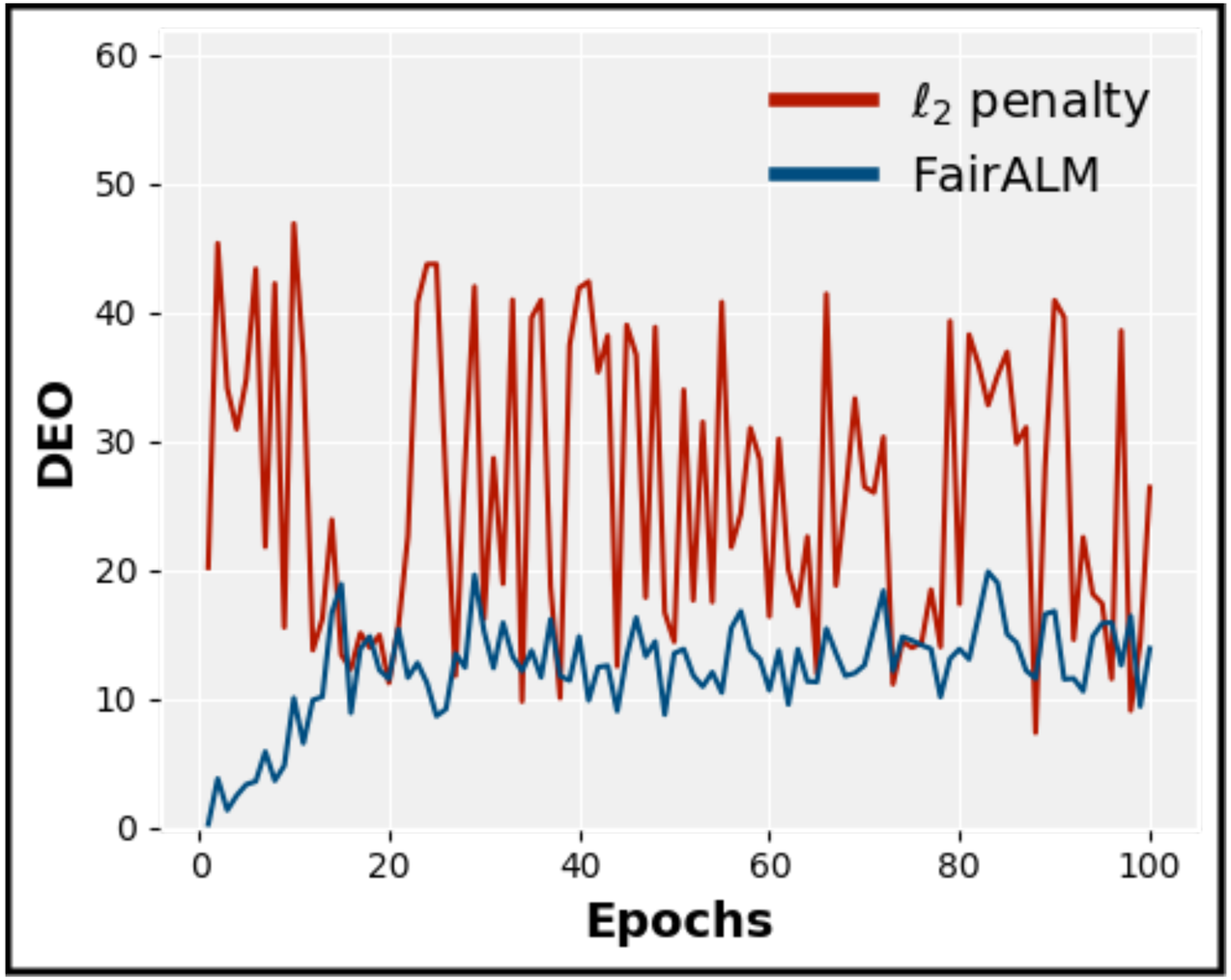
FairALM

1: $\lambda_0 = 0, \quad w_0 = 0$
2: **FOR** $t = 0,1,2,\ldots,T$ **DO**
3: **Sample** $z \sim$ **Data**
4: **Pick** $v_t \in \partial\left(\hat{e}_{h_w} + (\lambda_t + \eta)\hat{\mu}^{s_0}_{h_w} - (\lambda_t - \eta)\hat{\mu}^{s_1}_{h_w}\right)$
5: $w_t \leftarrow w_{t-1} - \tau v_t$
6: $\lambda \leftarrow \lambda_t + \eta(\hat{\mu}^{s_0}_{h_w} - \hat{\mu}^{s_1}_{h_w})$
7: **END**

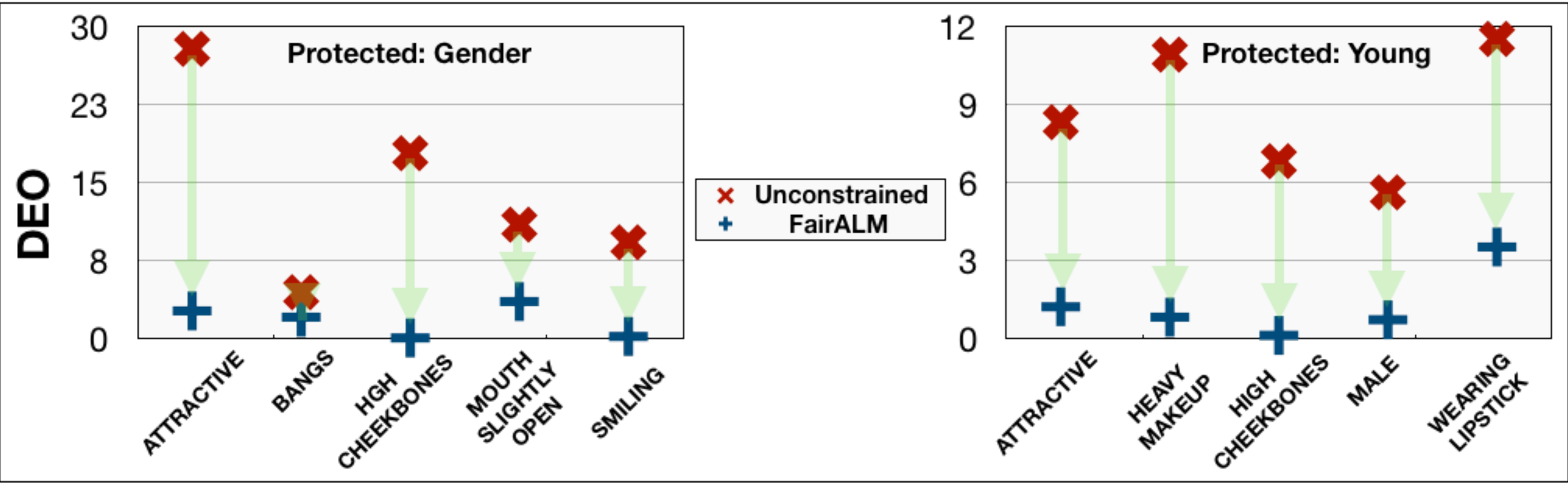
- Notations
- The Lagrangian Formulation
- FairALM
    - Linear Classifiers
    - Deep Networks
- Experimental Results

CELEBA DATASET

A STABLE TRAINING PROFILE

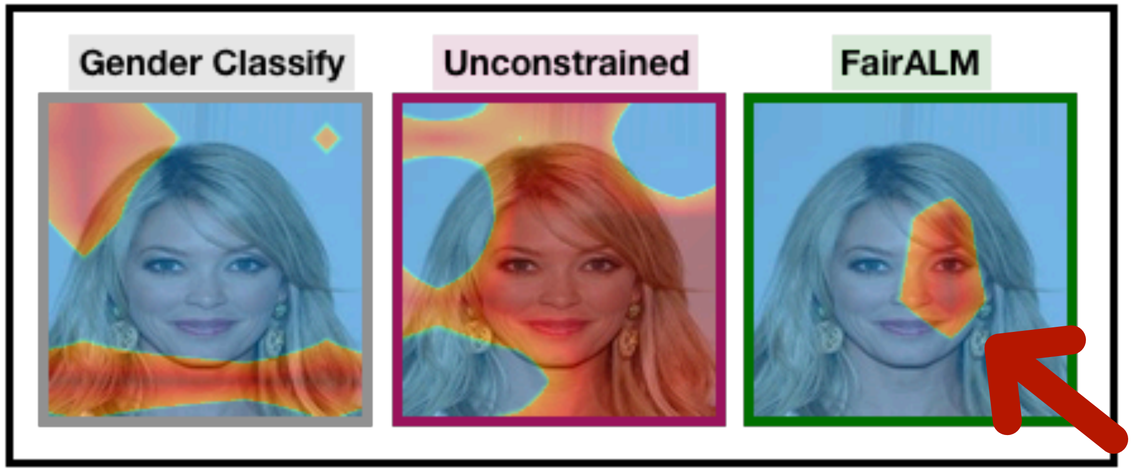


CELEBA DATASET



CELEBA DATASET

Interpretable Models



IMSITU DATASET

Cooking (+)
Driving (-)

Unconstrained

Focus: **PERSON**
Feature leaking Gender

FairALM

Focus: **FOOD**
Feature concealing Gender

IMSITU DATASET

Microwaving (+)
Pumping (-)

Unconstrained

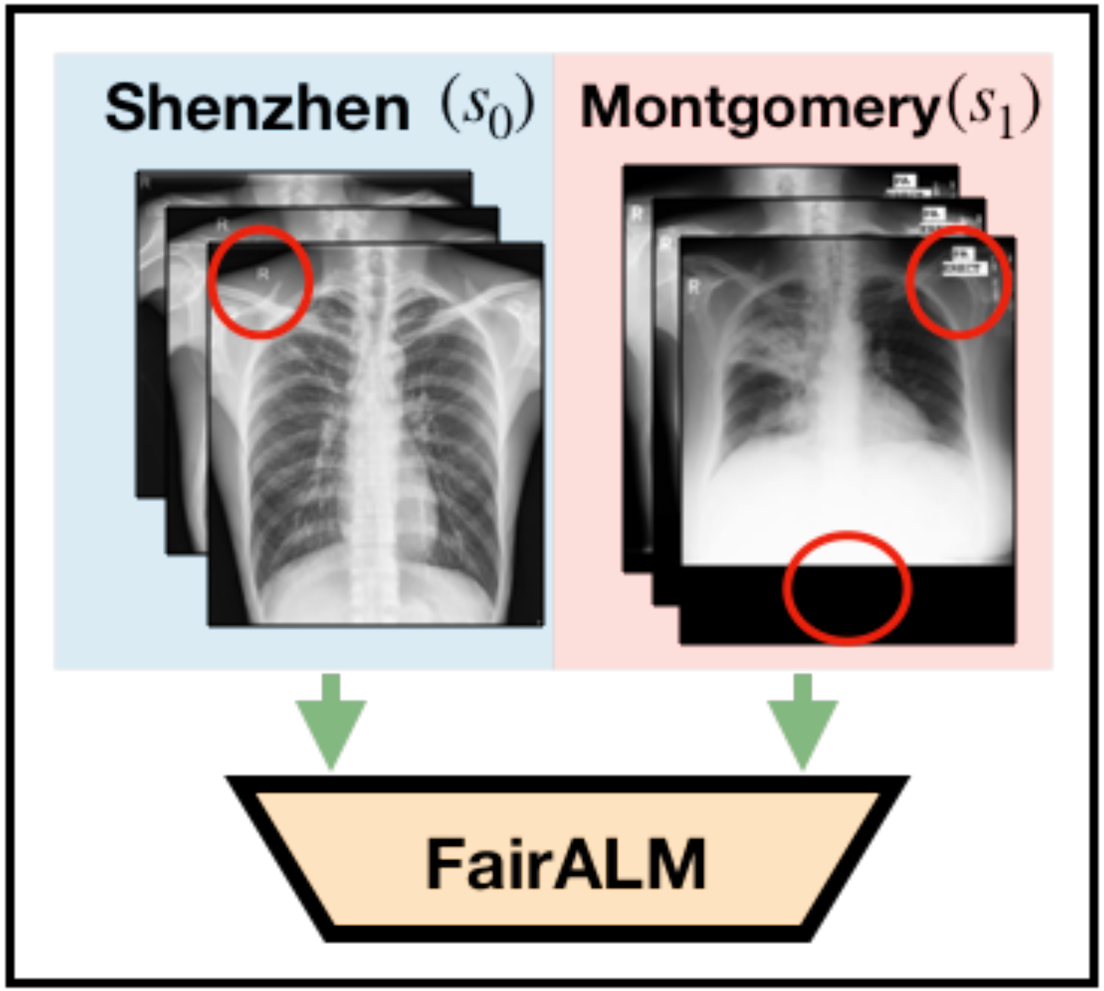
Focus: **FACE**
Feature revealing Gender

FairALM

Focus: **MICROWAVE/UTENSILS**
Feature concealing Gender

BEYOND FAIRNESS

MULTI-SITE POOLING



Ite Missa Est