Appendix

Reproducibility

Fairness Metrics

In this study, we use ABROCA [Gardner *et al.*, 2019] to evaluate the predictive fairness. As pointed out in [Gardner *et al.*, 2019], both positive and negative predictions may induce different impacts on students, e.g., being predicted to be at risk of dropping out or not may lead to different types or levels of support for students. Therefore, ABROCA measures the difference in overall accuracy across compared groups, i.e., the absolute value of the area between the ROC curves of the compared groups. Mathematically, ABROCA is computed from the predicted probabilities and true labels, i.e., $\int_0^1 |ROC_{g1}(t) - ROC_{g2}(t)|$, where $g1$ and $g2$ are the two groups compared, i.e., the privileged group and unprivileged group.

Training Details

Forum Post Classification. For the testbed, we first extract the text embedding of each post using BERT without fine-tuning, which would be then fed up as the input features to the Logistic Regression. For model training in all cases, we set the *batch size* as 32, and *learning rate* as 0.001. For all debiasing approaches, we select the best model according to the performance on the validation set, i.e., the model with minimum validation loss.

STEM Career Prediction. We adopted the model[1] open-sourced by the authors of the original paper [Yeung and Yeung, 2019].

Hyperparameter Tuning

Pre-processing approaches that require parameter tuning include `FairBalance`, `Preferential Sampling`, `Massaging`, and `Debiasing BERT`, we detailed their parameters tuning as below:

- For `FairBalance`, we selected the clustering algorithms following original paper from the list *kmeans*, *agglomerative clustering*, and *spectral clustering*.
- For the `Debiasing BERT` we set the max iteration of continue training as 12, and for each iteration we use the validation set as the reference set to determine the number of instances to be sampled for each subgroup.
- For `Preferential Sampling`, we set the ranker, i.e., the classifier to identify borderline instances, as the logistic regression.
- For `Massaging`, the ranker is same as `Preferential Sampling`, i.e., logistic regression.
- For `Correlation Remover`, we tune the parameter α from $[0.1, 0.2, 0.3, ..., 0.9, 1]$.

For in-processing approaches, we detailed the parameter settings as below:

- For `Reduction`, we select the best grid size from $\{10, 15, 20, 25, 30\}$ for each variant using parameterized with demographic parity and equalized odds.
- For `FairBatch` we follow the original paper to select α, i.e., the learning rate of FairBatch, from $[1e-4, 3e-4, 5e-4, 7e-4, 1e-3, .., 0.03, 0.05]$, for each variant using parameterized with demographic parity, equal opportunity, and equalized odds.
- For `Adversarial Debiasing`, we select the adversary weight within the range $[0.1, 1]$ with a step size as 0.1.

In terms of post-processing approaches, we detailed the parameter settings as below:

- For `CalibratedEqOdds`, we follow the original paper to select the error rate from *false positive rate*, *false negative rate*, and *weighted*.
- For `FairProjection`, we set the cross-entropy as the divergence function and follow the original paper to use mean equalized odds as the fairness constraint.

Implementation Details

All the methods were implemented and trained using Tensorflow. For each debiasing approach, we check if they were publicly available and open-sourced, if so we adapted the code into our evaluation framework. Specifically, for approaches `Reduction` and `CalibratedEqOdds`, we used the implementations provided by fairlearn[2] [Bird *et al.*, 2020] and aif360[3] [Bellamy *et al.*, 2019]. For re-sampling approach `SMOTE` we used the implementation from [Lemaître *et al.*, 2017].

References

[Bellamy *et al.*, 2019] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilović, et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4–1, 2019.

[Bird *et al.*, 2020] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach, and Kathleen Walker. Fairlearn: A toolkit for assessing and improving fairness in ai. *Microsoft, Tech. Rep. MSR-TR-2020-32*, 2020.

[Gardner *et al.*, 2019] Josh Gardner, Christopher Brooks, and Ryan Baker. Evaluating the fairness of predictive student models through slicing analysis. In *Proceedings of the 9th international conference on learning analytics & knowledge*, pages 225–234, 2019.

[Lemaître *et al.*, 2017] Guillaume Lemaître, Fernando Nogueira, and Christos K Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *The Journal of Machine Learning Research*, 18(1):559–563, 2017.

[1] https://github.com/ckyeungac/ADM2017

[2] https://fairlearn.org/v0.8/api_reference/fairlearn.reductions.html#fairlearn.reductions.GridSearch

[3] https://aif360.readthedocs.io/en/latest/modules/generated/aif360.sklearn.postprocessing.CalibratedEqualizedOdds.html

[Yeung and Yeung, 2019] Chun-Kit Yeung and Dit-Yan Yeung. Incorporating features learned by an enhanced deep knowledge tracing model for stem/non-stem job prediction. *International Journal of Artificial Intelligence in Education*, 29(3):317–341, 2019.