FairUP: A Framework for Fairness Analysis of Graph Neural Network-Based User Profiling Models

Mohamed Abdelrazek*
Otto von Guericke University Magdeburg
Magdeburg, Germany
mabdelrazek105@gmail.com

Erasmo Purificato*
Otto von Guericke University Magdeburg
Magdeburg, Germany
Leibniz Institute for Educational Media |
Georg Eckert Institute
Brunswick, Germany
erasmo.purificato@ovgu.de

Ludovico Boratto
Department of Mathematics and Computer Science,
University of Cagliari
Cagliari, Italy
ludovico.boratto@acm.org

Ernesto William De Luca
Otto von Guericke University Magdeburg
Magdeburg, Germany
Leibniz Institute for Educational Media |
Georg Eckert Institute
Brunswick, Germany
ernesto.deluca@ovgu.de

ABSTRACT

Modern user profiling approaches capture different forms of interactions with the data, from user-item to user-user relationships. Graph Neural Networks (GNNs) have become a natural way to model these behaviours and build efficient and effective user profiles. However, each GNN-based user profiling approach has its own way of processing information, thus creating heterogeneity that does not favour the benchmarking of these techniques. To overcome this issue, we present FairUP, a framework that standardises the input needed to run three state-of-the-art GNN-based models for user profiling tasks. Moreover, given the importance that algorithmic fairness is getting in the evaluation of machine learning systems, FairUP includes two additional components to (1) analyse pre-processing and post-processing fairness and (2) mitigate the potential presence of unfairness in the original datasets through three pre-processing debiasing techniques. The framework, while extensible in multiple directions, in its first version, allows the user to conduct experiments on four real-world datasets. The source code is available at https://link.erasmopurif.com/FairUP-source-code, and the web application is available at https://link.erasmopurif.com/FairUP.

CCS CONCEPTS

• **Human-centered computing → User models**; • **Social and professional topics → User characteristics**; • **Applied computing → Law, social and behavioral sciences**.

KEYWORDS

User Profiling, Graph Neural Networks, Algorithmic Fairness

ACM Reference Format:
Mohamed Abdelrazek, Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. 2023. FairUP: A Framework for Fairness Analysis of Graph Neural Network-Based User Profiling Models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3539618.3591814

1 INTRODUCTION

Recent years have seen a huge increase in personal information being shared through daily interaction with artificial intelligence (AI) systems, especially information retrieval (IR) systems and recommender systems (RSs). In this context, **user profiling** has emerged as an influential topic in various practical applications, notably in social media [23], e-commerce [31] and personalised systems [11]. User profiling aims to deduce an individual's preferences, characteristics or actions from gathered data, enabling the development of an effective *user model*. In particular, modern systems emphasise *implicit user profiling*, which entails analysing end-users actions and interactions to create user profiles. These techniques are also known as *behavioural user profiling* [19]. Graphs depict a natural means to model these behaviours by representing users as nodes and their interactions as edges. Graph representation learning approaches [16], principally **Graph Neural Networks** (GNNs) [15, 20, 29, 34, 35] provide a significant advantage over traditional deep learning (DL) methods to deal with such data. Although existing GNNs have been proven to be successful in user profiling tasks, whose effectiveness is generally assessed through the ability to correctly classify a user's attribute (e.g. salary, consumption level, age, etc.) [6], they are susceptible to learning biases present in the data and reflect them in their outputs, like any machine learning (ML) model that is trained on historical data.

*Both authors contributed equally to this research and appear in alphabetical order.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07.
https://doi.org/10.1145/3539618.3591814

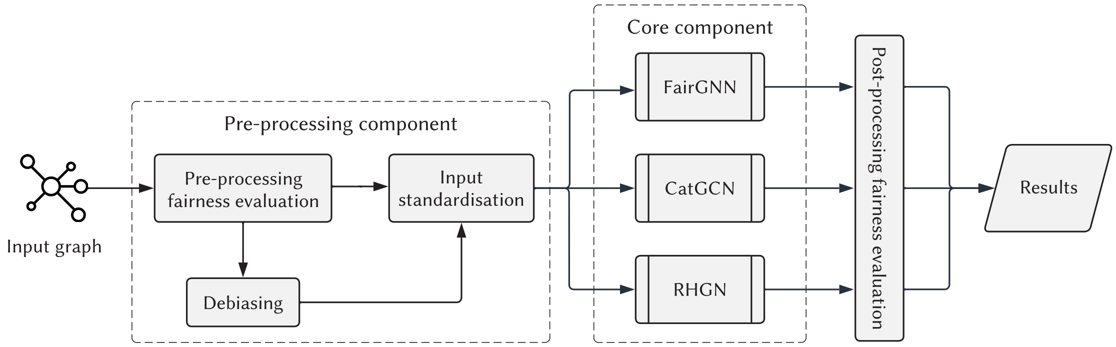

Figure 1: Logical architecture of the FairUP framework.

In such a scenario, and due to the social impact automated decision-making systems currently have on people's everyday life, **algorithmic fairness** [21, 24] has turned into a key research field in AI. Several works have been published in the last few years about bias detection and mitigation for general ML models [1, 4, 30], user-related scenarios [26], IR systems [12, 14, 27] and RSs [22, 28]. Instead, there are only a limited number of studies assessing the fairness of GNNs [7–9, 25], and to the best of our knowledge, no tool has yet been realised to allow a comprehensive and comparative fairness analysis for these architectures.

Our contributions. In this paper, we propose **FairUP**, a novel framework to assess algorithmic fairness on GNN-based models for user profiling tasks. The presented framework is founded on our previous and first-of-its-kind analysis of the fairness of state-of-the-art GNN-based behavioural user profiling models [25], published last year. FairUP empowers researchers and practitioners to simultaneously examine classification performance and fairness metrics scores of the included models. Specifically, it is composed of several components (described in detail in the continuation of the paper), which allow end users to:

(1) compute the fairness of the input dataset by means of a pre-processing metric, i.e. *disparate impact* [1];
(2) mitigate the unfairness of the dataset, if needed, by applying different well-established debiasing methods, i.e. *sampling* [18], *reweighting* [18] and *disparate impact remover* [13];
(3) standardise the input (a graph in *Neo4J*[1] or *NetworkX*[2] format) for each GNN model;
(4) train one or more GNNs, i.e. *CatGCN* [5], *RHGN* [32] and *FairGNN* [7], by specifying the parameters for each of them;
(5) evaluate post-hoc fairness by exploiting four standard metrics, i.e. *statistical parity* [10, 13], *equal opportunity* [17], *overall accuracy equality* [2] and *treatment equality* [2].

Our contributions can be summarised as follows:

- We designed an extensible and standardised framework capable of training state-of-the-art GNN-based user profiling models on different graph datasets, detecting pre-processing and post-processing bias, and mitigating potential unfairness in the original dataset. The source code is made available[3].
- We developed a user interface (UI) which enables users to explore the framework and examine its functionalities.
- We implemented a prototype version of the framework operating on pre-defined real-world datasets. The web application[4] and a demo video[5] are accessible online.

2 THE FAIRUP FRAMEWORK

In this section, we describe in detail the components of the proposed framework, whose logical architecture is shown in Figure 1.

2.1 Pre-processing component

The goal of this component is to properly pre-process the input data for the different models available in the framework. The pre-processing component itself is divided into three modules: the optional *pre-processing fairness evaluation* and *debiasing* followed by the *input standardisation*.

Pre-processing fairness evaluation. As discussed in Section 1, FairUP aims to support any input graph from the user, in either Neo4J or NetworkX format. Once the input data is correctly decoded, the user has the opportunity to measure the overall dataset fairness, and this is achieved by calculating the **disparate impact** metric [1]. The concept of disparate impact can also be depicted or used as a general fairness metric based on evaluating the proportion of individuals who receive a positive outcome depending on their sensitive attribute (e.g. age, country, gender, etc.). This metric describes covert, frequently unintentional discrimination when procedures or systems appear to treat people equally, and it is defined as:

$$\frac{P\,(Y=1 \mid D=unprivileged)}{P\,(Y=1 \mid D=privileged)} \tag{1}$$

According to the literature and industry standards [3], a violation of disparate impact occurs when the positive outcome given

[1] https://neo4j.com/.
[2] https://networkx.org/.
[3] **Source code** available at https://link.erasmopurif.com/FairUP-source-code.
[4] **Web application** available at https://link.erasmopurif.com/FairUP.
[5] **Demo video** available at https://link.erasmopurif.com/FairUP-demo-video.

to the unprivileged group is less than 80% of their proportional representation compared to the privileged group.

Debiasing module. After evaluating the dataset fairness, if biases are uncovered, the user can decide to apply a pre-processing debiasing approach. The framework supports three methods: *sampling*, *reweighting* and *disparate impact remover*.

Sampling [18] is an approach which attempts to re-sample the dataset in a way so that the discrimination is mitigated or removed. After partitioning the dataset into four groups (*deprived community with positive class labels*, *deprived community with negative class labels*, *favoured community with positive class labels* and *favoured community with negative class labels*), it calculates for each class label and sensitive attribute the expected sizes if the given dataset would have been non-discriminatory. Finally, a *uniform* or *preferential sampling* algorithm is applied.

Reweighting [18] tries to mitigate bias in the dataset by assigning different weights to the dataset tuples. In particular, by giving the unfavourable sensitive attribute higher weights than favourable ones. To achieve this, it calculates the *expected* and *observed* probability for a given sensitive attribute label and class label. If the expected probability is higher than the observed probability, there is a bias towards the opposite class label. To overcome that, lower weights are assigned to tuples that are favoured.

Disparate impact remover [13], as the name suggests, has been specifically developed to remove disparate impact bias from a dataset. This is done by editing the sensitive attribute features so that the correlation between those features and the prediction class is reduced and kept balanced for all prediction classes in the dataset. Disparate impact remover also ensures that while editing the sensitive attributes features the group ranking of the different prediction classes is preserved.

Input standardisation. Since each GNN model requires the input to be structured in a specific manner, this module transforms the original dataset as expected by the selected models. Being the framework extensible, when a new GNN is added, the related input standardisation procedure must be implemented.

2.2 Core component

This part represents the core of the proposed framework. In its first version, we included three state-of-the-art GNN-based models, which demonstrated the most effective in user profiling: *CatGCN*, *RHGN* and *FairGNN*.

CatGCN [5] is a graph convolutional network designed for graph learning on categorical node features. It improves the initial representation of nodes by incorporating two forms of explicit interaction modelling into its learning process, a local interaction based on multiplying node feature pairs and a global interaction based on adding features to an artificial graph. The results demonstrate the efficacy of performing feature interaction modelling before graph convolution.

RHGN [32] is a Relation-aware Heterogeneous Graph Network developed to model the various relationships present in a heterogeneous graph that encompasses different entities. At its core lies a multi-relation attention mechanism that resembles a transformer, used to evaluate the significance of nodes and meta-relations in the graph. Furthermore, a heterogeneous graph propagation network is utilised to gather information from numerous sources. This method has been proven to produce better results than several graph neural network-based models in tasks related to user profiling.

FairGNN [7] is a GNN framework tailored for mitigating bias in model predictions using an in-processing debiasing approach. The framework is divided into three models: a classifier (which can be either a Graph Convolutional Network (GCN) [33] or a Graph Attention Network (GAT) [29]), a sensitive attribute estimator and an adversary. This model uses a min-max game approach to mitigate bias between the classifier and the adversary, where the adversary tries to estimate sensitive attributes in the dataset from the predicted node representation by the classifier, and the classifier aims to learn node representations that fool the adversary to make wrong predictions.

2.3 Post-processing fairness evaluation

After the chosen models are trained, the framework can compute the fairness of their predictions, by exploiting four standard metrics: *statistical parity*, *equal opportunity*, *overall accuracy equality*, and *treatment equality*.

The specific formulas are defined considering $y \in \{0, 1\}$ as the binary target label and $\hat{y} \in \{0, 1\}$ as the prediction of the user profiling model $f : x \longrightarrow y$, while the sensitive attribute is indicated with $s \in \{0, 1\}$.

Statistical parity [10, 13], also known as *demographic parity*, is a measure of fairness in which each group has an equal chance of being designated as the positive class, meaning that predictions independent with sensitive attributes.

$$SP = P(\hat{y} = 1|s = 0) = P(\hat{y} = 1|s = 1) \tag{2}$$

Equal opportunity [17] demands the probability of a member of a positive class to be assigned to the positive class should be equal for each group.

$$EOD = P(\hat{y} = 1|y = 1, s = 0) = P(\hat{y} = 1|y = 1, s = 1) \tag{3}$$

Overall accuracy equality citeberk2021overallaccuracy characterises fairness as the equal probability of a subject from either positive or negative class to be assigned to its respective class, i.e. each group should have the same prediction accuracy.

$$\begin{aligned} OAE &= P(\hat{y} = 0|y = 0, s = 0) + P(\hat{y} = 1|y = 1, s = 0) \\ &= P(\hat{y} = 0|y = 0, s = 1) + P(\hat{y} = 1|y = 1, s = 1) \end{aligned} \tag{4}$$

Treatment equality [2] requires the error rate produced by the classifier to be the same across different sensitive groups

$$TED = \frac{P(\hat{y} = 1|y = 0, s = 0)}{P(\hat{y} = 0|y = 1, s = 0)} = \frac{P(\hat{y} = 1|y = 0, s = 1)}{P(\hat{y} = 0|y = 1, s = 1)} \tag{5}$$

To be able to integrate such metrics in our framework, we quantitatively evaluate the fairness of our predictions using the presented metrics according to our previous work [25]:

$$\Delta_{SP} = |P(\hat{y} = 1|s = 0) - P(\hat{y} = 1|s = 1)| \tag{6}$$

$$\Delta_{EOD} = |P(\hat{y} = 1|y = 1, s = 0) - P(\hat{y} = 1|y = 1, s = 1)| \tag{7}$$

$$\begin{aligned} \Delta_{OAE} = |&P(\hat{y} = 0|y = 0, s = 0) + P(\hat{y} = 1|y = 1, s = 0) \\ &-P(\hat{y} = 0|y = 0, s = 1) + P(\hat{y} = 1|y = 1, s = 1)| \end{aligned} \tag{8}$$

$$\Delta_{TED} = \left| \frac{P(\hat{y} = 1|y = 0, s = 0)}{P(\hat{y} = 0|y = 1, s = 0)} - \frac{P(\hat{y} = 1|y = 0, s = 1)}{P(\hat{y} = 0|y = 1, s = 1)} \right| \tag{9}$$

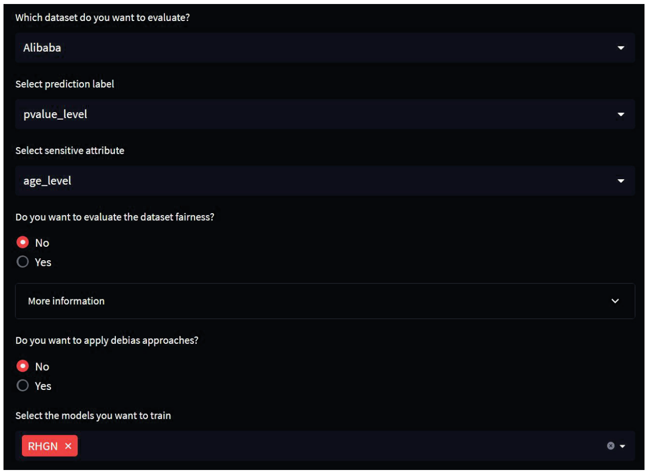

Figure 2: Selection of the dataset, input parameters and pre-processing fairness functionalities.

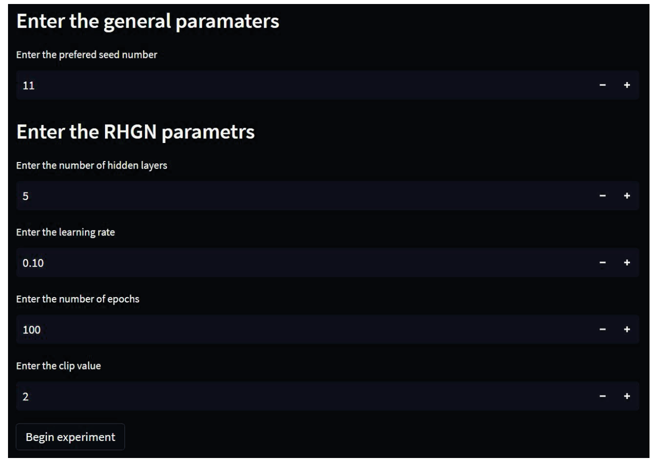

Figure 3: Selection of the training parameters for the chosen GNN model(s). In this example, RHGN parameters are shown.

3 DEMONSTRATION

In this section, we guide our readers in the exploration of the functionalities of the FairUP UI, developed to allow users to interact with the framework. The UI is developed using the Python library named *Streamlit*[6].

The initial page displays the description of the framework components, as well as the architecture. Upon clicking on the "*Framework*" button in the navigation sidebar, the user will be directed to the main page (Figure 2), where it is possible to select the input dataset, the related target class and sensitive attribute to adopt in the experiment (from a dropdown menu which appears once a dataset is selected) and decide whether or not to execute the pre-processing fairness functionalities (illustrated in Section 2.1).

To allow quick use of FairUP, the users have the possibility to select a *preset* configuration for each dataset instead of manually choosing every attribute.

For our first prototype version of the framework, four pre-defined datasets are made available: *Alibaba*, *JD*, *NBA* and *Pokec-z*.

Alibaba[7] is a public large-scale e-commerce user profiling dataset that contains about 350 000 users with information about their purchases and click rates. The dataset is based on the Chinese Alibaba portal and was used in [32] and [5] for the original experiments.

JD[8] is the second dataset originally adopted in [32] and [5] for training and evaluation. The dataset consists of users and items from the JD e-commerce platform and contains information about click and purchase relationships between them.

NBA[9] is one of the dataset used in [7]. It contains information about around 400 NBA players from the 2016-2017 season, such as player height, player salary, player team and age. The dataset was further extended by the paper authors to obtain graphs that link NBA players together, leveraging data crawled from Twitter.

[6]https://streamlit.io/
[7]https://tianchi.aliyun.com/dataset/56
[8]https://github.com/guyulongcs/IJCAI2019HG
[9]https://www.kaggle.com/datasets/noahgift/social-power-nba

Pokec-z[10] is the second dataset utilised in [7]. Pokec is a social network dataset that contains millions of users. Its data is similar to Facebook and Twitter content, with information about users' gender, age, working field, etc. The dataset also includes user mutual relationships, which makes it a good social network dataset for GNNs training.

As discussed, the user has the flexibility in the number of models to be trained. After the choice has been taken, the framework requires inserting the training parameters for each model (Figure 3). At the end of this procedure, the user can start the experiment and check a table reporting the classification and fairness results.

4 CONCLUSION

In this work, we presented FairUP, a novel framework for fairness analysis and mitigation of GNN-based user profiling models. The framework enables users to easily analyse the different GNN model prediction results as well as the fairness metrics scores of these predictions. Nonetheless, the framework also allows users to mitigate the bias in the predictions by applying different pre-processing debiasing approaches before training. Finally, we presented a simple UI that facilitates the user to interact with such a complex framework structure effortlessly. FairUP offers end-users the opportunity to understand and compare different GNN models and also examine various bias detection and mitigation approaches using a standardised tool. In the future, the framework can be extended to support new models and fairness procedures, i.e. metrics for multi-classification problems. Additionally, it will support different debiasing approaches, which can include in-processing and post-processing techniques.

REFERENCES

[1] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning*. fairmlbook.org. http://www.fairmlbook.org.
[2] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2021. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research* 50, 1 (2021), 3–44.

[10]https://snap.stanford.edu/data/soc-pokec.html

[3] Dan Biddle. 2017. *Adverse impact and test validation: A practitioner's guide to valid and defensible employment testing.* Routledge.

[4] Simon Caton and Christian Haas. 2020. Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053* (2020).

[5] Weijian Chen, Fuli Feng, Qifan Wang, Xiangnan He, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. CatGCN: Graph Convolutional Networks with Categorical Node Features. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[6] Weijian Chen, Yulong Gu, Zhaochun Ren, Xiangnan He, Hongtao Xie, Tong Guo, Dawei Yin, and Yongdong Zhang. 2019. Semi-supervised user profiling with heterogeneous graph attention networks. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence.* 2116–2122.

[7] Enyan Dai and Suhang Wang. 2021. Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. In *Proceed. of the 14th ACM International Conference on Web Search and Data Mining.* 680–688.

[8] Yushun Dong, Jian Kang, Hanghang Tong, and Jundong Li. 2021. Individual fairness for graph neural networks: A ranking based approach. In *Proceed. of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining.* 300–310.

[9] Yushun Dong, Ninghao Liu, Brian Jalaian, and Jundong Li. 2022. Edits: Modeling and mitigating data bias for graph neural networks. In *Proceedings of the ACM Web Conference 2022.* 1259–1269.

[10] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference.* 214–226.

[11] Christopher Ifeanyi Eke, Azah Anir Norman, Liyana Shuib, and Henry Friday Nweke. 2019. A survey of user profiling: State-of-the-art, challenges, and solutions. *IEEE Access* 7 (2019), 144907–144924.

[12] Michael D Ekstrand, Anubrata Das, Robin Burke, Fernando Diaz, et al. 2022. Fairness in information access systems. *Foundations and Trends® in Information Retrieval* 16, 1-2 (2022), 1–177.

[13] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 259–268.

[14] Ruoyuan Gao and Chirag Shah. 2019. How fair can we go: Detecting the boundaries of fairness optimization in information retrieval. In *Proceedings of the 2019 ACM SIGIR international conference on theory of information retrieval.* 229–236.

[15] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[16] William L Hamilton. 2020. Graph representation learning. *Synthesis Lectures on Artifical Intelligence and Machine Learning* 14, 3 (2020), 1–159.

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

[18] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems* 33, 1 (2012), 1–33.

[19] Sumitkumar Kanoje, Sheetal Girase, and Debajyoti Mukhopadhyay. 2015. User profiling trends, techniques and applications. *arXiv preprint arXiv:1503.07474* (2015).

[20] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings.*

[21] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ashesh Rambachan. 2018. Algorithmic fairness. In *Aea papers and proceedings*, Vol. 108. 22–27.

[22] Jurek Leonhardt, Avishek Anand, and Megha Khosla. 2018. User fairness in recommender systems. In *Companion Proc. of The Web Conference 2018.* 101–102.

[23] Lizi Liao, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2018. Attributed social network embedding. *IEEE Transactions on Knowledge and Data Engineering* 30, 12 (2018), 2257–2270.

[24] Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. 2021. Algorithmic fairness: Choices, assumptions, and definitions. *Annual Review of Statistics and Its Application* 8 (2021), 141–163.

[25] Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. 2022. Do Graph Neural Networks Build Fair User Models? Assessing Disparate Impact and Mistreatment in Behavioural User Profiling. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management.* 4399–4403.

[26] Erasmo Purificato, Flavio Lorenzo, Francesca Fallucchi, and Ernesto William De Luca. 2022. The Use of Responsible Artificial Intelligence Techniques in the Context of Loan Approval Processes. *International Journal of Human–Computer Interaction* (2022), 1–20.

[27] Guilherme Ramos and Ludovico Boratto. 2020. Reputation (In)dependence in Ranking Systems: Demographics Influence Over Output Disparities. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020.* ACM, 2061–2064. https://doi.org/10.1145/3397271.3401278

[28] Guilherme Ramos, Ludovico Boratto, and Carlos Caleiro. 2020. On the negative impact of social influence in recommender systems: A study of bribery in collaborative hybrid algorithms. *Inf. Process. Manag.* 57, 2 (2020), 102058. https://doi.org/10.1016/j.ipm.2019.102058

[29] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[30] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *IEEE/ACM International Workshop on Software Fairness (FairWare 2018).* IEEE, 1–7.

[31] Chuhan Wu, Fangzhao Wu, Junxin Liu, Shaojian He, Yongfeng Huang, and Xing Xie. 2019. Neural demographic prediction using search query. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining.* 654–662.

[32] Qilong Yan, Yufeng Zhang, Qiang Liu, Shu Wu, and Liang Wang. 2021. Relation-aware Heterogeneous Graph for User Profiling. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management.* Association for Computing Machinery, New York, NY, USA, 3573–3577.

[33] Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 7370–7377.

[34] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. 2019. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* 793–803.

[35] Ziwei Zhang, Peng Cui, and Wenwu Zhu. 2022. Deep Learning on Graphs: A Survey. *IEEE Transactions on Knowledge and Data Engineering* 34, 1 (2022), 249–270.