UNFair: Search Engine Manipulation, Undetectable by Amortized Inequity

TIM DE JONGE, Radboud University Nijmegen, The Netherlands
DJOERD HIEMSTRA, Radboud University Nijmegen, The Netherlands

Modern society increasingly relies on Information Retrieval systems to answer various information needs. Since this impacts society in many ways, there has been a great deal of work to ensure the fairness of these systems, and to prevent societal harms. There is a prevalent risk of failing to model the entire system, where nefarious actors can produce harm outside the scope of fairness metrics. We demonstrate the practical possibility of this risk through UNFair, a ranking system that achieves performance and measured fairness competitive with current state-of-the-art, while simultaneously being manipulative in setup. UNFair demonstrates how adhering to a fairness metric, Amortized Equity, can be insufficient to prevent Search Engine Manipulation. This possibility of manipulation bypassing a fairness metric discourages imposing a fairness metric ahead of time, and motivates instead a more holistic approach to fairness assessments.

CCS Concepts: • **Information systems** → **Learning to rank**.

Additional Key Words and Phrases: Fairness, Information Retrieval, Search Engine Manipulation Effect, Exposure, UNFair

ACM Reference Format:
Tim de Jonge and Djoerd Hiemstra. 2023. UNFair: Search Engine Manipulation, Undetectable by Amortized Inequity. In . ACM, New York, NY, USA, 10 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Modern society increasingly relies on Information Retrieval systems to answer various information needs, ranging from high impact applications like healthcare [16] and automated fact checking [39] to more everyday problems such as fashion matching [46] and music recommendation [25]. Since Information Retrieval systems impact society in many ways, there has been a great deal of work to ensure the fairness of these systems. While it is easy to see why automated fact checking requires some care, nearly all Information Retrieval systems have some fairness concerns, and there is a broad spectrum of efforts to increase fairness in these systems [8, 22].

While these efforts are all for the greater good, many authors do not cite the harms they aim to address [5, 15]. This has the risk of fairness interventions being deployed in situations where they do not mitigate the harm they ought to address, or where a fairness metric is kept in use although it does not adequately measure the harm in the situation.

Selbst et al. provide an excellent overview of difficulties in abstracting the complicated notion of fairness, by indicating five traps that fair-ML work falls into. First and foremost of these is the Framing Trap, or "[The] failure to model the entire system over which a social criterion, such as fairness, will be enforced". This means an algorithm can appear fair to a model, while the algorithm causes societal harm outside the scope of assessment. [44].

FAccT 2023, June 12–15, 2023, Chicago, IL, USA
2023. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

We demonstrate the dangers of the Framing Trap through UNFair, a novel ranking system that achieves performance and measured fairness competitive with current state-of-the-art, while being clearly manipulative in setup. UNFair demonstrates how failing to model the entire system can result in Search Engine Manipulation while adhering to a specific fairness metric: Amortized Equity [4]. If manipulating Amortized Inequity is possible, imposing a fairness metric ahead of time might be insufficient to prevent societal harm, which motivates a more holistic approach to fairness assessments [1, 26].

In Section 2, we will introduce Search Engine Manipulation as a harm in Information Retrieval. In section 3, we detail the model of the Information Retrieval system we investigate in this work. In section 4, we look at Amortized Equity, a common fairness metric, and introduce our own metric to assess Search Engine Manipulation. In section 5, we introduce our own ranking system, *UNFair*, to illustrate how ranking systems could bypass carelessly applied fairness metrics. In section 6, we show our experimental setup and the results thereof. In section 7, we discuss our findings, and provide an outlook for research in Fair Ranking.

2 SEARCH ENGINE MANIPULATION

As illustration of the societal harms that can result from Information Retrieval systems, we use the Search Engine Manipulation Effect. Epstein and Robertson [13] held an experiment in which they showed participants the results of a mock search engine, manipulated to be heavily politically skewed; their experimental design can be found in Figure 1. First, they polled their participants on a prospective election. They then asked the participants to interact with the search engine results page that was biased towards one of the candidates. They then polled the participants again, to see whether a shift had taken place. The results varied strongly with participants' features, and their prior knowledge of the election. The net probability of influencing a vote in the direction of the manipulation could be as high as 25% to the most susceptible subgroup of the population: poorly informed, moderate voters [13]. By contrast, the least susceptible subgroup showed a slight negative response, making it less likely for the participant to vote for the candidate favoured by the search results.

The manipulation of search results can be difficult to detect for end-users. Since each user only interacts with the search engine a limited amount of times, it can be hard to find patterns in the results provided by a search engine. Additionally, we cannot expect users to have the expertise required to detect bias in search results if they use a search engine to acquire knowledge in the first place. The powerful position in which large tech companies find themselves, combined with the inability for individuals to realize something is afoot, makes for a nasty problem if manipulation through search engines does take place.

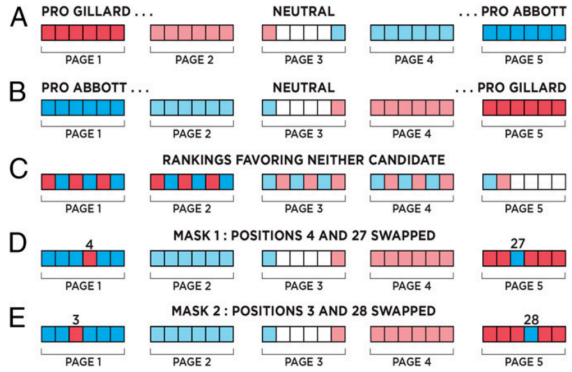

Fig. 1. Experimental design of the Search Engine Manipulation Effect, taken from [13]. Condition C is a neutral condition, with conditions A and B skew towards Left and Right respectively. Conditions D and E are altered versions of condition B to mask the bias in the results.

Although there are no direct examples of Search Engine Manipulation being used with malicious intent, there are adjacent cases. In the 2012 US Election, the Obama administration cooperated narrowly with Google to ensure electoral advantage [47]. Google could leverage their swathes of personal data to estimate "persuasion scores"; they could then target advertisements towards those individuals that were easiest to persuade, to ensure maximal impact. Similarly, in the 2016 US Election, the Trump administration narrowly cooperated with Cambridge Analytica on Facebook to micro-target advertisements specifically to those most susceptible to them [3] – Cambridge Analytica claims to have done the same in several other elections around the globe [38]. While these examples are not necessarily Search Engine Manipulation - and their effectiveness unclear [11, 17, 20, 36] - they show that these tech giants are moving in political spheres with the intent of granting electoral advantage.

3 MODEL SPECIFICATION

3.1 Learning-to-Rank

We must first formalize an abstraction of an Information Retrieval system to investigate how societal harm may result from Information Retrieval systems. Table 1 contains an overview of the notation used in this paper.

The core task of an Information Retrieval system is to rank a set of documents $\mathcal{D} = \{d_i\}$ optimally given a query. Primarily, we want to show users those documents they want to see, but as section 4 will show, other conditions (diversity of results, non-discrimination, fairness) can be imposed as well. We use an online Learning-to-Rank model, which entails that at the start of a session, the model does not have any information on how much the population is interested in each document: it will have to learn by doing [23]. Learning-to-Rank is common in Information Retrieval, where frequently corpora are too large to rely on expert judgements. Instead, Learning-to-Rank algorithms are used to dynamically ensure that users of a model are shown good results, even with relatively sparse feedback.

We build our Information Retrieval system in the model used in Morik et al. [27]. Following Morik et al. [27], we observe a political news site. Assume the editors of the site have selected 30 documents d_i to be displayed to users, and the Learning-to-Rank algorithm must rank these documents by estimated relevance, using the data users provide: in this case, clicks. Sequentially, users $x_t \in \mathcal{X}$ come to the site, and are presented with a ranking of documents. Looking through these documents, the user will give attention $a^t_{d_i}$ to documents, or alternatively phrased, the documents get exposure. As a user is exposed to these documents, they will click on whichever documents they are interested in. These clicks are registered by the model as $c^t_{d_i}$. The ranking system's target is then to interpret these clicks to produce the best ranking according to some quality standard; relevance, fairness, or any other definition.

Morik et al. [27] introduces two rankers: *D-ULTR(Glob)*, standing for Dynamic-Unbiased Learning-To-Rank(Global) and FairCo. We will now take a look at D-ULTR(Glob); we take a closer look at FairCo in section 5.1. Both FairCo and UNFair are extensions of this D-ULTR(Glob) system.

D-ULTR(Glob) mitigates a bias users have, where they interact with documents because they are at the top of the ranking, not because the documents are especially relevant to them [19]. This means we cannot use click data as a direct proxy for relevance: the users clicked more on documents that were near the top of the ranking than these documents deserved. If we assume an attention distribution, we can use Inverse Propensity Scoring to calculate the estimated relevance $\hat{r}^t_{d_i}$ from the users' clicks [30, 43]. D-ULTR(Glob) then ranks the documents by $\hat{r}^t_{d_i}$ [27]. We use D-ULTR(Glob) as a baseline that is not concerned with fairness issues.

We illustrate D-ULTR(Glob) with a simple example: A user enters the site, clicks on the articles in position 1, 5, and 18, and leaves the site. The site registers these clicks, and passes them on for further processing. Since the user clicked on the articles on position 1, 5, and 18, we assume the user was interested in these articles, and we put article 1, 5, and 18 at the top of the results page for the next user. Since users are most likely to scroll through results from the first to the last, the user is much more likely to interact with article 1 than with article 18. For the next user, Inverse Propensity Score then makes article 18 the new article 1, article 5 the new article 2, and article 1 the new article 3. We do not have any information on any of the other documents, so they remain in the order they were. After enough iterations of additive feedback, the model will converge to a ranking that has a high probability of producing relevant documents for any given user joining the site.

3.2 Technical Specification

In this section, we lay out the details of the model the ranking system will operate in.

At time t, a user x_t visits the page, and requires a ranking of the documents $d_i \in \mathcal{D}$. The user-base is made up of two smaller sub-populations. The left-wing sub-population is drawn from a normal distribution centered around -0.5, and analogously, the right-wing sub-population is drawn from a normal distribution centered around 0.5. We truncate the distributions of users to $[-1, 1]$ for ease of graphing; this does not result in any substantial changes to the results of any experiments. The portion of left-wing users is represented by p_{left}. In this model, we assume that everyone is either left-wing or right-wing, so the portion of right-wing users is given by $1 - p_{\text{left}}$. Combining these concepts, the political lean of users is given by

$$p(x_t) \sim p_{\text{left}}\mathcal{N}(-0.5, 0.2) + (1 - p_{\text{left}})\mathcal{N}(0.5, 0.2)$$

where $p(x_t)$ is restricted to $[-1, 1]$. Here, a political lean of -1 means that the user is maximally left wing, a political lean of 1 means that the user is maximally right-wing. The documents' political lean are normally distributed around 0; $p(d_i) \sim \mathcal{N}(0, 0.5)$, similarly restricted to $[-1, 1]$. The phrasing here implies a two-party system; this need not be the case. In pluralistic political system one can run this same analysis by substituting left-wing with the party of one's choice, and right-wing by a fitting comparison group. Section 4.2 motivates the choice for only two groups, and gives an indication of how these groups ought to be chosen.

We describe by users' *openness* their willingness to interact with documents that do not align with their political preferences. The users' openness is uniformly distributed between 0.05 and 0.55, $\sigma(x_t) \sim \mathcal{U}(0.05, 0.55)$. We also interpret users' openness as their *susceptibility*: their likelihood to change their vote based on the results page.

These factors interact in a natural way. A user is more likely to find a document relevant if the political lean of the document is close to the political lean of the user; the more open a user is, the more receptive they are for documents that do not align with their own political preference. Operationalizing the probability of user x_t finding document d_i relevant, we have

$$R(x_t, d_i) = e^{-\frac{(p(x_t) - p(d_i))^2}{2\sigma(x_t)^2}}$$

The actual relevance of document d_i to user x_t is then 1 with probability $R(x_t, d_i)$, and 0 otherwise:

$$r^t_{d_i} \sim \text{Bernoulli}(p = R(x_t, d_i))$$

Unfortunately, outside of a laboratorium setting, we cannot directly observe relevance; rather, we are restricted to observing user interactions with documents. For a user to interact with a document, they must both find the document relevant, and have seen it in the first place. We assume the attention a user gives an article is dependent on the position a document gets in the ranking; the higher the document ranks, the more likely it is for a user to see it [19]. We denote the rank document d_i gets for user x_t under system π as $\pi(d_i, x_t)$. We conceptualize the probability that user x_t sees document d_i as the *attention* or *exposure* user x_t gives document d_i, and calculated as

$$a^t_{d_i} = \frac{1}{\log_2(\pi(d_i, x_t) + 1)}$$

Note that, indeed, attention has the range $[0, 1]$, meaning interpreting it as a probability is consistent. The attention function indicates that the top-rated document is very likely to be seen, with a rapid initial drop. This drop then levels out after the first few results, meaning there is less difference between the documents on rank 20 and rank 21 than there is between the documents on rank 2 and rank 3.

As before, we require a binary value for a user seeing a document, leading to the following variable for a user seeing a document: $s^t_{d_i} \sim \text{Bernoulli}(a^t_{d_i})$. Finally, a user clicks on the document if and only if they have seen the document and they deem it relevant; $c^t_{d_i} = s^t_{d_i} a^t_{d_i}$. The model can only observe click feedback: it does not have access to any of the aforementioned underlying variables. As an exception, we assume that UNFair has access to the susceptibility of the users, using it only to ensure the manipulation described in section 5.2.

Again, let us illustrate this model with an example. A user Robin comes to the page; they are from the left-wing portion of the population. Robin's political lean $p(x_{\text{Robin}})$ is 0.4, and their openness $\sigma(x_{\text{Robin}})$ is 0.3. They come across a document from ProPublica d_{PP}, which has a political lean $p(d_{\text{PP}}) = 0.2$. The probability that Robin finds the document d_{PP} relevant is then $R(x_{\text{Robin}}, d_{\text{PP}}) = e^{-\frac{(0.4-0.2)^2}{2*0.3^2}}$, or about 80%.

Here, Robin found the document on position 3 ($\pi(d_{\text{PP}}, x_{\text{Robin}}) = 3$). The odds of Robin seeing the ProPublica article are then $a^{\text{Robin}}_{d_{\text{PP}}} = \frac{1}{\log_2(3+1)}$, or 50%. Now, Robin is a person rather than a concept, so we can actually be sure whether they see the article, and whether they find it relevant. It turns out that they do find the article relevant, and that they see it. They click on it, and the system registers Robin's click on the ProPublica article.

Variable	Notation
Documents	$\mathcal{D} = \{d_i\}$
Users	$\mathcal{X} = \{x_t\}$
Attention	$a^t_{d_i}$
Relevance (merit)	$r^t_{d_i}$
Estimated Relevance	$\hat{r}^t_{d_i}$
Political lean (user)	$p(x_t)$
Political lean (document)	$p(d_i)$
Groups	G_-, G_+
Clicks	$c^t_{d_i}$
Document Position	$\pi(d_i, x_t)$
Susceptibility (openness)	σ_t
Model Performance wrt Relevance	nDCG@$k(t)$

Table 1. Notations used in this paper, and a short representation of their meanings.

4 EVALUATION

4.1 Relevance

We discern three ways of evaluating an Information Retrieval system.

Primarily, an Information Retrieval system ought to provide users with those documents they deem relevant. To assess relevance, we use nDCG@10 [18]; the technical specification of nDCG can be found in Appendix A. Although delivering users the articles they want to see is an important interpretation of performance, it is not the sole possible interpretation; we should also give documents fair treatment. As such, we assess ranking systems on fairness, through Amortized Inequity. The details of Amortized Inequity can be found in the subsequent section 4.2. Finally, to assess Electoral Impact, we introduce a metric to measure specifically Search Engine Manipulation as introduced in Section 2. Section 4.3 provides the details of this metric.

4.2 Fairness

Assessing whether any system is fair is not a trivial undertaking. If users click on document A slightly more than they click on document B, it seems sensible that the system shows document A slightly more. If users only have a minimal preference for document A, but the system always ranks document A higher than document B, this seems to violate proportionality. The main thing to determine in assessing a system is then the appropriate level to which there can be differentiation between documents, before this difference in attention becomes harmful.

To this end, we divide the documents into subgroups $G_n \subset \mathcal{D}$ which partition the set of documents. Additionally, we conceptualize merit $m^t_{d_i}$ as the extent of differentiation that is appropriate - if we treat each group according to their merit, this is fair, and the further we stray from treatment according to merit, the more unfair it becomes. This notion of treating each group according to merit is called Equity of Attention. If we have more than two groups, we can calculate the Multi-lateral Inequity as follows:

$$\text{Multi-lateral Inequity} = \sum_{i=1}^{n} \sum_{j=i}^{n} \frac{\sum_{d \in G_i} a_d}{\sum_{d \in G_i} m_d} - \frac{\sum_{d \in G_j} a_d}{\sum_{d \in G_j} m_d}$$

With Multi-lateral Inequity, we compare between all groups, and then average out the deviations [4, 27, 31]. As mentioned by Biega et al. [4], Multi-lateral Inequity is primarily useful as an optimization target: when unfairness is zero, all groups are treated equitably according to the chosen definition of merit. As a metric of the level of skew in the system, Multi-lateral Inequity is less convenient. It is impossible to see *which* deviations from the average are taking place since all pairwise differences are summed. The case where all groups are scattered closely around the mean is much different in analysis than the case in which one group is heavily discriminated against, and one group heavily favoured. To ease analysis, we suggest using a bi-lateral fairness definition for measuring inequity [41]. In so doing, one has to take care to appropriately choose the groups tested; one group G_- for which one suspects unfair inequality and an appropriate comparison group G_+. In the case with two groups, we denoted inequity as follows:

$$\text{Bi-lateral Inequity} = \frac{\sum_{d \in G_+} a_d}{\sum_{d \in G_+} m_d} - \frac{\sum_{d \in G_-} a_d}{\sum_{d \in G_-} m_d} \tag{1}$$

We can attach different meanings to merit, dependent on circumstance. More restrictive than Equity of Attention is Equality of Attention: a special case where we give each group equal merit. Equality of Attention can be a good choice in cases where we cannot accurately measure or estimate a document's value, but it is too restrictive to use much otherwise. More frequently, merit is attached to relevance: show a document to users proportional to the extent that users want to see it.

Relevance is conceptually attractive, but noisy to the point of being practically intractable. This means that any assessment of real-world Information Retrieval systems on relevance will have to either estimate relevance from user behaviour, or create a more labour intensive inquiry, in which users or judges have to investigate all documents and mark down the extent they found each document relevant to their query. TREC, the largest applied Text Retrieval conference, historically used only binary relevance, which lacks nuance compares to the different ways in which users might experience documents in the real world. Work to comprehensively define relevance in the best way to assess Information Retrieval systems in general is ongoing, and has proven to be quite challenging [24, 35, 40].

To combat the noise in relevance, we can assess fairness across multiple queries. Here, we accumulate relevance and attention across the queries that the system has seen, and only compare the totals against each other, through Amortized Inequity.

$$\text{Amortized Inequity} = \frac{\sum_{x_t \in \mathcal{X}} \sum_{d \in G_+} a^t_d}{\sum_{x_t \in \mathcal{X}} \sum_{d \in G_+} m^t_d} - \frac{\sum_{x_t \in \mathcal{X}} \sum_{d \in G_-} a^t_d}{\sum_{x \in \mathcal{X}} \sum_{d \in G_-} m^t_d} \tag{2}$$

Biega et al. [4] introduced the term Amortized Inequity, but the accumulation of inequity across queries is commonplace, either through direct summation over visited queries [9, 10, 27, 31, 33, 42, 45], or an expected value over query space [29, 34]. Amortized inequity can then be used as a target; as long as the amortized inequity is (close to) zero, both groups have received attention in proportion to their merit.

4.3 Electoral Impact

As suggested in Section 2, it might be possible to build a system that adheres to the restrictions posed by fairness metrics, while resulting in societal harm. To assess whether Search Engine Manipulation takes place, we devise the following measure:

$$\text{Amortized Impact} = \sum_{x_t \in \mathcal{X}} \sigma_t \left(\sum_{d \in \mathcal{G}_+} a^t_d - \sum_{d \in \mathcal{G}_-} a^t_d \right)$$

Here, we assume every document has impact on the reader x_t equal to the attention a^t_d the document received; the reader is then impacted proportional to their susceptibility σ_t. As mentioned in Section 2, Google and Facebook have both attempted to estimate users' susceptibility in previous political contexts, with the explicit intention of influencing elections.

Amortized Impact is a continuous extension of the theory established in Epstein and Robertson [13]. Systems can manipulate people's electoral preference by up to 25% with heavily skewed search results pages, which matches the range of Amortized Impact. To account for queries that are less heavily skewed, Amortized Impact scales the impact by the difference in attention the groups receive.

We acknowledge that Amortized Impact makes simplifying assumptions. It has embedded in itself the disputed claims that one can assess ahead of time how much a user's vote will be swayed by a search engine [28], that every results page has a non-zero effect on a user's vote [11], or the more fundamental complaint that the likelihood of someone's vote might not be desirable or possible to capture in data.

Even with these limitations, there is value in examining this metric as a proxy for real-world impact; most model analysis stops at the model borders, whereas value can be found in observing the layer beyond the model output [44]. Other choices can be made on the exact specification of amortized impact - the following discussion still holds.

5 FAIR RANKERS

5.1 FairCo

FairCo [27] is a Learning-to-Rank system that targets Amortized Inequity as fairness metric. Since this paper has received the Best Paper award at SIGIR 2020 and is a general-purpose ranker, we will use it as state-of-the-art baseline.

To understand FairCo, first recall from Section 3.1 how D-ULTR(Glob) operates. D-ULTR(Glob) estimates the relevance of a document, corrected for position bias; we denote this $\hat{r}^t_{d_i}$. D-ULTR(Glob) then creates a results page by ranking documents by this estimated relevance. To ensure that FairCo does not allow for any inequity of attention, Morik et al. modify this base ranker. To do so, they use Amortized Inequity (equation 1). Amortized Inequity is transformed into an error term $\mathbf{err}_i$: $\mathbf{err}_i = 0$ whenever d_i is in the privileged group, and $\mathbf{err}_i$ is equal to the Amortized Inequity otherwise. Ranking by $\hat{r}^t_{d_i} + \mathbf{err}_i$ then produces a ranking system that converges in the limit to having zero Amortized Inequity.

5.2 UNFair

We set out to demonstrate that a ranking system can result in societal harm, while adhering to the restrictions imposed by a fairness metric. To this end, we create the Undetectably Nefarious Fair Ranker, or UNFair. It is important to understand that our intention is not to contribute to the state-of-the-art of electoral manipulation. Rather, the possibility of the UNFair algorithm existing motivates a careful use of fairness metrics: careless application of fairness metrics might not stop the harm it intends to stop. The code to run UNFair can be found on https://github.com/UNFairPaper/UNFair.

UNFair is a system with 3 conditions: Baseline, Positive, and Negative. UNFair assigns one of these conditions to each user, and ranks the documents based on the condition the user is in. In the Baseline condition, UNFair uses an appropriate high-performing ranker; in this case, we use D-ULTR(Glob) as introduced in section 3.1 [27]. The Baseline ranker does not need to be fair; UNFair solves for the fairness metric through use of the Positive and Negative conditions.

The Positive condition is a direct modification of the Baseline condition. Here, UNFair first ranks all documents according to the Baseline ranker. Then, the Positive ranker finds the first k documents with positive political lean ($p(d_i) > 0$), and places them on the first k positions of the results page, pushing other results down as necessary. This results page is then presented to the user. The Positive condition corresponds to case A in Figure 1. The Negative condition is defined similarly, but replacing all instances of 'positive' with 'negative', and corresponding to case B in Figure 1.

UNFair assigns conditions to users by susceptibility score σ_t: UNFair offers the Positive condition to those users with the highest susceptibility scores, and the Negative ranker to those with the lowest susceptibility scores. Herein lies the manipulation: since not all attention is equal, the algorithm aims to have all high-impact attention go towards the privileged group, while all low impact attention goes towards the maltreated group. This means that the voters that can be influenced, will be influenced towards the positive group. There is formal equity of attention, while the system can produce electoral impact.

To formalize this, we keep thresholds σ_- and σ_+: if $\sigma_t > \sigma_+$, UNFair uses the Positive condition, likewise if $\sigma_t < \sigma_-$, UNFair uses the Negative condition. If $\sigma_- \leq \sigma_t \leq \sigma_+$, UNFair uses the Baseline condition.

Recall that we are working inside a Learning-to-Rank model, which means that at the start of an experiment, we do not know anything about the user distribution. This means we have to update σ_- and σ_+ throughout a run, to maintain balance between the different conditions.

UNFairSmooth is the first mechanism of choosing thresholds: here, the thresholds move smoothly within the domain. After every timestep, UNFairSmooth sets $\sigma_- = \sigma_- + \lambda \text{err}_t$, and similarly $\sigma_+ = \sigma_+ + \lambda \text{err}_t$. This ensures that UNFairSmooth chooses the negative condition more frequently when the discrepancy in exposure is positive, and vice versa. We investigate the results of choosing the parameter λ in Section 6.

UNFairCoarse is the other mechanism of choosing thresholds: UNFairCoarse uses only a limited set of possible threshold values. The consequence of this is that we can execute more explicit control over the model and ensure convergence, at the cost of some flexibility. For convenience, we use the notation $corr_t$ = Amortized Inequity $* t$, where t is the amount of users. If we can ensure that $|corr_t| < k$ for some k, we guarantee that Amortized Inequity remains below $\frac{k}{t}$, and as such, that it tends to 0.

To ensure that $|corr_t| < k$, divide the population into evenly sized chunks along the susceptibility axis. Then, when the system is in balance, the system only uses the Left and Right ranker for the outermost 2%, and uses the baseline condition otherwise. If $corr_t > 10$, i.e. there is an inequity of $\frac{10}{t}$ towards the positive, the Negative ranker takes over the lowest susceptibility chunk of the population. As $corr_t$ rises, UNFairCoarse will assign more chunks of population to the Negative condition, thus stopping the rise of $corr_t$. This will result in Amortized Inequity tending to 0 as previously explained.

6 RESULTS

6.1 Experimental setup

In the following section, we will demonstrate that UNFair can attain similar performance and fairness to FairCo, while reaching substantial electoral impact. The code to reproduce these results can be found on https://github.com/UNFairPaper/UNFair. Unless otherwise listed, we set FairCo's λ parameter to 0.01 as suggested in Morik et al. [27], UNFair's λ parameter to 0.005, and p_{left} to 0.5. Each experiment uses 30 random seeds, that have not been previously used in developing UNFair, or in previous experiments. Within each experiment, we then evaluate all systems on the same random seeds, resulting in the same experimental conditions.

6.2 Can UNFair achieve Electoral Impact while maintaining competitive performance and fairness?

Figure 2 shows a long run of these systems, evaluated on relevance, fairness, and electoral impact. For this experimental setup, we Learn-to-Rank the same 30 documents for 10000 simulated users. In figure 2, the classical trade-off between fairness and performance is clearly visible [12]. D-ULTR(Glob) performs best in terms of Relevance, but is unfair according to Exposure Inequity. UNFairCoarse and FairCo have very similar performance on Relevance and Fairness, achieving 0 inequity with the same performance loss. Furthermore, we can see that FairCo reduces electoral impact to 0 as well, and that D-ULTR(Glob) does not stray far from neutral electoral impact. Both implementations of UNFair do deviate from 0, and in UNFairSmooth we can see oscillating behaviour as more users interact with the system. This oscillating behaviour is due to the delay between an imbalance on Amortized Equity, and UNFairSmooth's response. UNFairSmooth's thresholds move in the opposite direction of the error term $\mathbf{err}_t$. If $\mathbf{err}_t$ is positive, but already decreasing, UNFairSmooth will still change the thresholds to be more skewed towards the negative side, thus overshooting the target. This results in spring-like behavior, as Amortized Equity will turn negative, rather than finding equilibrium at 0. UNFair's λ parameter can be interpreted as a spring constant; we explore the effects of varying λ in section 6.4.

6.3 How does performance vary at different levels of evaluation?

For completeness, we investigated how the Relevance Assessment changes as we vary the amount of documents evaluated. In figure 3, we can see the differences between the different evaluations. UNFairSmooth performs well on nDCG@1, but UNFairSmooth's performance declines as more results are evaluated. FairCo and UNFairCoarse perform about equally well throughout, but always worse than D-ULTR(Glob).

It is notable that nDCG@30 is much higher than nDCG@5, for all rankers. This is not reflective of the rankers being more proficient at ranking all documents. Instead, this is a quirk of the model chosen: the performance of a purely random ranker is also significantly higher when evaluated with nDCG@30. As such, comparisons between systems on the same metric are valid, comparisons between metrics mostly are not.

6.4 What are the effects of UNFair's Lambda parameter?

In the following experiment, we only vary λ in UNFairSmooth, without comparison against a baseline. Figure 4 shows similar performance between all rankers. $\lambda = 0.05$ performs poorly, while $\lambda = 0.003$ performs best. A higher λ results in heavier oscillation in Electoral Impact, where Exposure Inequity is initially driven down but does not continue a steady decline. Conversely, a low λ results in steady, but slow behaviour. These observations are consistent with the interpretation of λ as spring constant of a system. As a middle ground between these two extremes, we retain $\lambda = 0.005$ for the other experiments.

6.5 Is UNFair effective at different user distributions?

Finally, it is worth observing what happens when populations shift. In figure 5, we vary the left-wing proportion of the population (p_{left}) between 35% and 65%, in intervals of 3%. The same four systems used before are evaluated here, on 30 previously unseen seeds. Plotted are the values the system produces after 3000 users have used the system.

In figure 5, in terms of nDCG, both UNFairSmooth and UNFairCoarse closely mimic FairCo's performance in terms of relevance, although all are lower than the baseline. We see here that FairCo slightly outperforms UNFair on minimizing inequity. UNFair does still achieve a serious reduction in inequity compared to D-ULTR(Glob). In terms of electoral impact, D-ULTR(Glob) has the most polarized scores. While FairCo mitigates the impact of the population skew, UNFair does not. In cases where the system is already skewed towards the positive side, UNFair does not reduce the impact much; in cases where the system is balanced, or skewed towards the negative side, UNFair moves this impact even further towards the positive side than FairCo does.

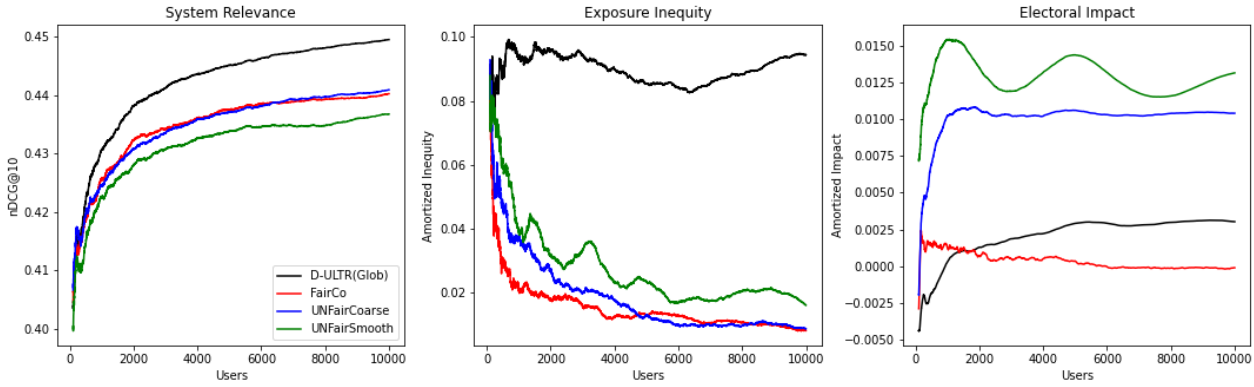

Fig. 2. nDCG@10 (left), Amortized Inequity (middle) and Amortized Impact (right), averaged over 30 trials of 10000 simulated users.

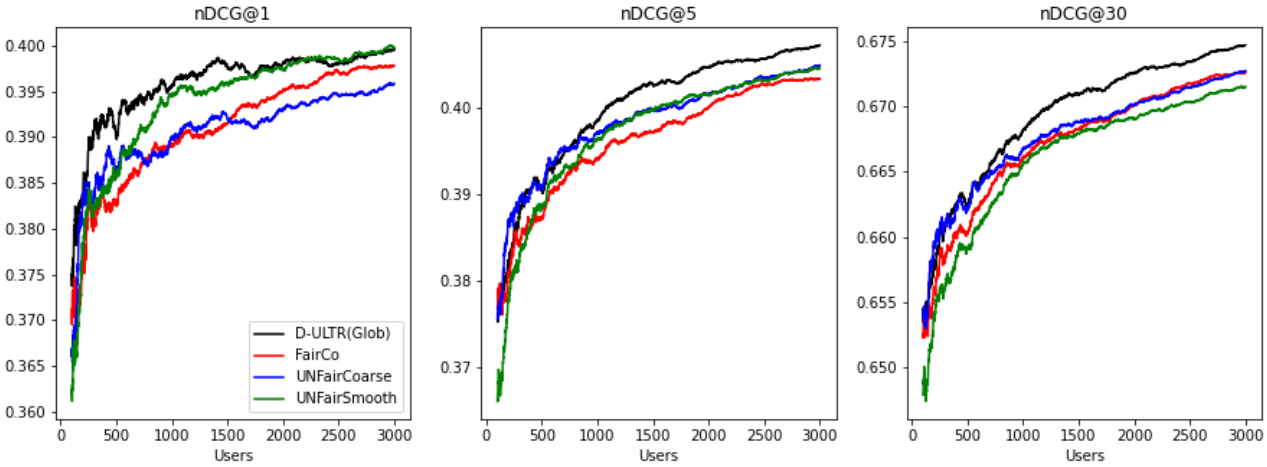

Fig. 3. From left to right: nDCG@1, nDCG@5, nDCG@30, averaged over 30 trials of 3000 simulated users

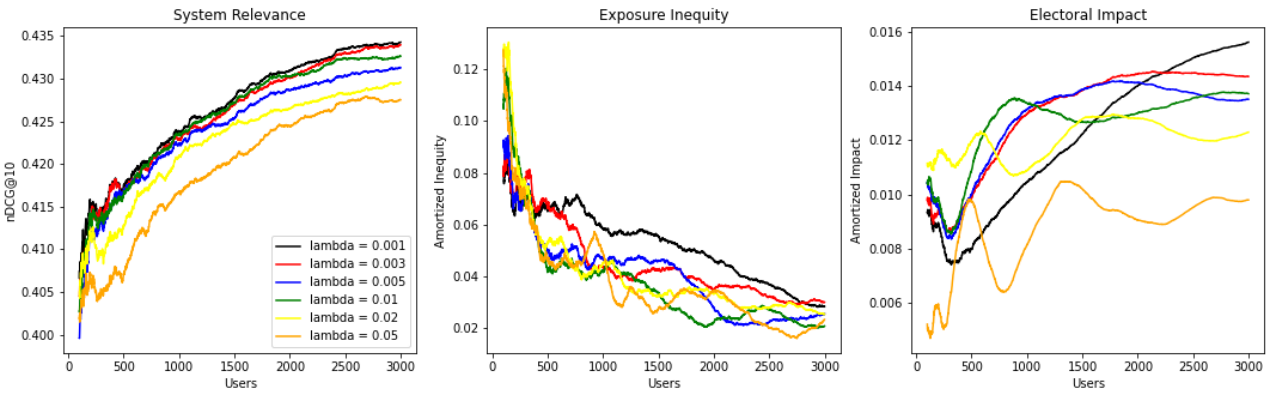

Fig. 4. nDCG@10 (left), Amortized Inequity (middle) and Amortized Impact (right) of UNFairSmooth with $\lambda \in [0.001, 0.003, 0.005, 0.01, 0.02, 0.05]$, averaged over 30 trials of 3000 users.

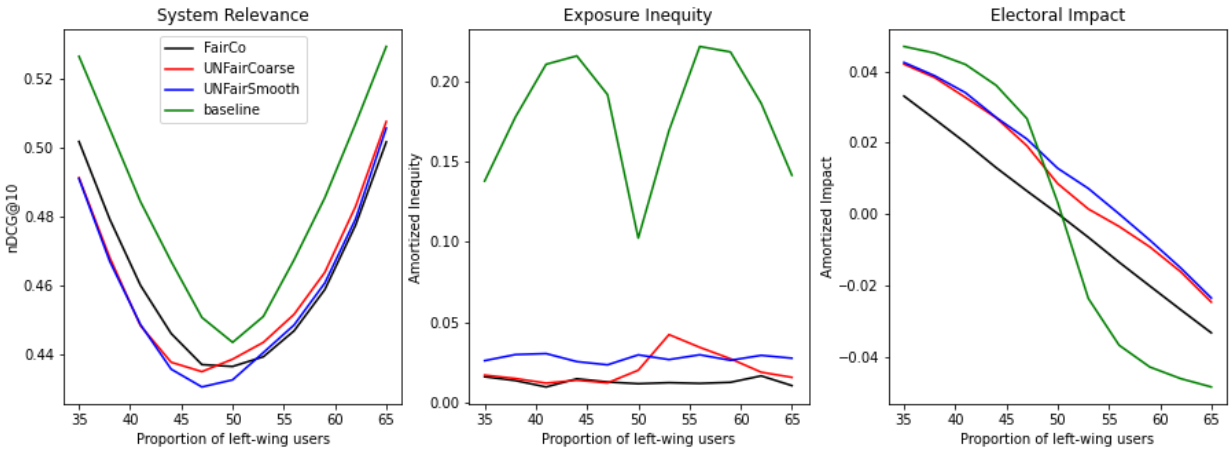

Fig. 5. nDCG@10 (left), Amortized Inequity (middle) and Amortized Impact (right), averaged over 30 trials of 3000 simulated users for each value of p_{left} between 35% and 65%.

7 DISCUSSION AND MOVING FORWARD

Throughout the results, UNFair reaches performance close to FairCo, while being similarly fair according to Amortized Inequity. While achieving that, UNFair does not mitigate any of the electoral impact D-ULTR(Glob) has as baseline. and when reasonable, UNFair adds a positive electoral impact.

That said, UNFair certainly has imperfections for practical use; one could create a more refined unfair ranker. However, the point of this paper is not to provide a new state-of-the-art for manipulating elections. The point is to illustrate that current fairness metrics need careful application. Although Amortized Inequity is a plausible metric for Search Engine Manipulation, we see that a nefarious actor could create a system that appears fair to Amortized Inequity, while not reducing search engine manipulation in any way.

The UNFair algorithm is a start towards investigating the limitations of fairness metrics in Information Retrieval, but it leaves some questions yet unanswered. This paper only demonstrates UNFair on one simulated dataset, on one metric. While the algorithm is agnostic to the metric used and the dataset, future research will have to point out whether UNFair functions equally well in other circumstances. In models that classify rather than rank, we have theorems proving the impossibility of satisfying multiple fairness metrics at the same time [2, 6, 21]. Similar theorems for ranking would be a very powerful tool to understand the trade-offs inherent in Fair Ranking.

Scholars have been pondering justice (or, fairness) for centuries. Given that definitive answers on justice are few and far between, distilling fairness down to a single number seems overzealous [1, 32]. Rather, any practical application of fairness algorithms needs to be grounded in the relevant literature outside of just Information Retrieval [5, 14], and supported by the normative assumptions that clarify which substantive fairness we are benefiting [7].

It appears that fairness metrics have a very fitting role as canary in the coal mine; once the canary falls to the bottom of the cage, it is time to evacuate the mine. However, a live canary does not mean the mine is in any way safe, and building policy to keep the canary safe is a fool's errand.

Just because there are limitations to the usage of fairness metrics, that does not mean the practice has to be abolished [37]. Energy invested towards the greater good is laudable, and the problem remains very complex. The suggestion here is that different people working towards 'the greater good' could have very different goals, and making these goals explicit allows us to judge systems more adequately [15]. In naively optimizing towards a chosen metric, one could miss their target of reducing societal harm, and be left with a system that only looks better to the metric.

A more holistic algorithmic design and assessment might be necessary to ensure fair algorithms. Mökander et al. [26] suggest Ethics Based Auditing as a method to ensure morally sustainable algorithms. Ethics Based Auditing expands on the auditing process of trying to assess the system from within the system: First, one identifies moral values and potential harms, so there can be continuous evaluation of whether the system upholds these moral values. This enables us to evaluate whether performance indicators used still measure the underlying norm, rather than only evaluating whether the system still adheres to the performance indicators.

8 CONCLUSION

We investigated Amortized Equity of Attention as definition of fairness, and showed how a system could manipulate electoral results while being fair to Amortized Equity. The proposed algorithm UNFair achieves competitive performance for a Fair Learning-To-Rank Information Retrieval system, while simultaneously being clearly manipulative in setup.

9 ACKNOWLEDGMENTS

We would like to thank Frederik Zuiderveen Borgesius, Marvin van Bekkum, Richard Felius, and the ADM Club for their valuable input throughout this research. This work is funded in part by the EU project OpenWebSearch.eu under GA 101070014.

10 ETHICAL CONCERNS

This paper establishes an algorithm to circumvent fairness measures to still achieve electoral manipulation. This method only becomes relevant if adhering to Amortized Equity were to become mandatory, without further qualititative measures. We deem it unlikely that this work will be used for nefarious purposes.

A RELEVANCE ASSESSMENT

To start, we can see whether users are shown documents that are relevant to their interests; this would be performance of a model in a classical sense. To assess performance according to relevance, we use the normalized Discounted Cumulative Gain, or nDCG [18]. nDCG is a derived metric from DCG, which is defined by

$$\mathrm{DCG}_t(\pi) = \sum_{d_i \in \mathcal{D}} \frac{r^t_{d_i}}{\log(\pi(d_i, x_t) + 1)}$$

Here, $\pi(d_i, x_t)$ is the position of document d_i for user x_t, when ranked under policy π. DCG in itself captures the intuition that relevant documents need to be placed near the top of a ranking, but since the range of DCG can vary between queries, comparing systems across queries is cumbersome.

To ease comparison across queries, DCG is normalized by dividing by the ideal DCG obtainable on the query. We evaluate the ideal ranking policy π^*, which orders all documents by relevance. The ideal ranker is not practically obtainable, but provides a useful benchmark. We denote the ideal DCG as $\mathrm{iDCG}_t = \mathrm{DCG}_t(\pi^*)$ and the normalized DCG (nDCG) as $\mathrm{nDCG}_t(\pi) = \frac{\mathrm{DCG}_t(\pi)}{\mathrm{iDCG}_t}$

As a result, nDCG is always between 0 and 1, allowing for comparison across queries. One nuance here is that it might not be appropriate to assume the user observes all documents presented. The DCG formula can be adapted to meet this need. We index the documents based on their position on the results page, DCG evaluated at the k^{th} position is calculated as $\mathrm{DCG@}k(t) = \sum_{d \in \mathcal{D}: \pi(d_i, x_t) < k} \frac{r^t_{d_i}}{\log(\pi(d_i, x_t)+1))}$ and nDCG can similarly be adapted to

$$\mathrm{nDCG@}k(t) = \frac{\mathrm{DCG@}k(t)}{\mathrm{iDCG@}k(t)}$$

As a result, we can tune our performance measure to observations made; if users typically investigate no more than 10 documents, nDCG@10 is suitable to assess whether any given system is delivering relevant results to the users. If we vary k, the order of preference of ranking systems can also vary.

REFERENCES

[1] Agathe Balayn and Seda Gürses. 2021. Beyond Debiasing: Regulating AI and its inequalities. *EDRi Report. https://edri. org/wp-content/uploads/2021/09/EDRi_Beyond-Debiasing-Report_Online. pdf* (2021).

[2] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning: Limitations and Opportunities*. fairmlbook.org. http://www.fairmlbook.org.

[3] Hal Berghel. 2018. Malice domestic: The Cambridge analytica dystopia. *Computer* 51, 5 (2018), 84–89.

[4] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*. 405–414.

[5] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 5454–5476. https://doi.org/10.18653/v1/2020.acl-main.485

[6] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* 5, 2 (2017), 153–163.

[7] A Feder Cooper and Ellen Abrams. 2021. Emergent unfairness in algorithmic fairness-accuracy trade-off research. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 46–54.

[8] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. 2022. A Survey of Research on Fair Recommender Systems. *arXiv preprint arXiv:2205.11127* (2022).

[9] Fernando Diaz, Bhaskar Mitra, Michael D Ekstrand, Asia J Biega, and Ben Carterette. 2020. Evaluating stochastic rankings with expected exposure. In *Proceedings of the 29th ACM international conference on information & knowledge management*. 275–284.

[10] Virginie Do, Sam Corbett-Davies, Jamal Atif, and Nicolas Usunier. 2022. Online certification of preference-based fairness for personalized recommender systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 6532–6540.

[11] Tim Draws, Nava Tintarev, Ujwal Gadiraju, Alessandro Bozzon, and Benjamin Timmermans. 2021. This Is Not What We Ordered: Exploring Why Biased Search Result Rankings Affect User Attitudes on Debated Topics. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 295–305.

[12] Sanghamitra Dutta, Dennis Wei, Hazar Yueksel, Pin-Yu Chen, Sijia Liu, and Kush Varshney. 2020. Is there a trade-off between fairness and accuracy? a perspective using mismatched hypothesis testing. In *International Conference on Machine Learning*. PMLR, 2803–2813.

[13] Robert Epstein and Ronald E Robertson. 2015. The search engine manipulation effect (SEME) and its possible impact on the outcomes of elections. *Proceedings of the National Academy of Sciences* 112, 33 (2015), E4512–E4521.

[14] Sina Fazelpour and Zachary C Lipton. 2020. Algorithmic fairness from a non-ideal perspective. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 57–63.

[15] Ben Green. 2019. "Good" isn't good enough. In *Proceedings of the AI for Social Good workshop at NeurIPS*, Vol. 17.

[16] Anat Hashavit, Hongning Wang, Raz Lin, Tamar Stern, and Sarit Kraus. 2021. Understanding and Mitigating Bias in Online Health Search. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 265–274. https://doi.org/10.1145/3404835.3462930

[17] Alex Hern. 2018. Cambridge Analytica: how did it turn clicks into votes. *The Guardian* 6 (2018).

[18] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[19] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Transactions on Information Systems (TOIS)* 25, 2 (2007), 7–es.

[20] Joshua L Kalla and David E Broockman. 2018. The minimal persuasive effects of campaign contact in general elections: Evidence from 49 field experiments. *American Political Science Review* 112, 1 (2018), 148–166.

[21] Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2016. Inherent Trade-Offs in the Fair Determination of Risk Scores. *CoRR* abs/1609.05807 (2016). arXiv:1609.05807 http://arxiv.org/abs/1609.05807

[22] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, Juntao Tan, Shuchang Liu, and Yongfeng Zhang. 2022. Fairness in Recommendation: A Survey. *arXiv preprint arXiv:2205.13619* (2022).

[23] Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval* 3, 3 (2009), 225–331.

[24] David E Losada, Javier Parapar, and Alvaro Barreiro. 2019. When to stop making relevance judgments? A study of stopping methods for building information retrieval test collections. *Journal of the Association for Information Science and Technology* 70, 1 (2019), 49–60.

[25] Alessandro B Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. 2021. Investigating gender fairness of recommendation algorithms in the music domain. *Information Processing & Management* 58, 5 (2021), 102666.

[26] Jakob Mökander, Jessica Morley, Mariarosaria Taddeo, and Luciano Floridi. 2021. Ethics-based auditing of automated decision-making systems: nature, scope, and limitations. *Science and Engineering Ethics* 27, 4 (2021), 1–30.

[27] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 429–438.

[28] Arvind Narayanan. 2019. How to recognize AI snake oil. *Arthur Miller Lecture on Science and Ethics* (2019).

[29] Harrie Oosterhuis. 2021. Computationally efficient optimization of plackett-luce ranking models for relevance and fairness. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1023–1032.
[30] Zohreh Ovaisi, Kathryn Vasilaky, and Elena Zheleva. 2021. Propensity-Independent Bias Recovery in Offline Learning-to-Rank Systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1763–1767.
[31] Amifa Raj, Connor Wood, Ananda Montoly, and Michael D Ekstrand. 2020. Comparing fair ranking metrics. *arXiv preprint arXiv:2009.01311* (2020).
[32] Andrew D Selbst, Danah Boyd, Sorelle A Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and abstraction in sociotechnical systems. In *Proceedings of the conference on fairness, accountability, and transparency*. 59–68.
[33] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2219–2228.
[34] Ashudeep Singh and Thorsten Joachims. 2019. Policy learning for fairness in ranking. *Advances in Neural Information Processing Systems* 32 (2019).
[35] Eero Sormunen. 2002. Liberal relevance criteria of TREC- counting on negligible documents?. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. 324–330.
[36] David Sumpter. 2018. Why the Facebook data available to Cambridge Analytica could not be used to target personalities in the US Presidential election. (2018). https://soccermatics.medium.com/why-the-facebook-data-available-to-cambridge-analytica-could-not-be-used-to-target-personalities-in-2904fa0571bd
[37] Jared Sylvester and Edward Raff. 2018. What about applied fairness? *arXiv preprint arXiv:1806.05250* (2018).
[38] Channel 4 News Investigations Team. 2018. Exposed: Undercover secrets of Trump's data firm. (2018). https://www.channel4.com/news/exposed-undercover-secrets-of-donald-trump-data-firm-cambridge-analytica
[39] Nguyen Vo and Kyumin Lee. 2019. Learning from Fact-Checkers: Analysis and Generation of Fact-Checking Language. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 335–344. https://doi.org/10.1145/3331184.3331248
[40] Ellen M Voorhees, Ian Soboroff, and Jimmy Lin. 2022. Can Old TREC Collections Reliably Evaluate Modern Neural Retrieval Models? *arXiv preprint arXiv:2201.11086* (2022).
[41] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2021. Why fairness cannot be automated: Bridging the gap between EU non-discrimination law and AI. *Computer Law & Security Review* 41 (2021), 105567.
[42] Lequn Wang, Yiwei Bai, Wen Sun, and Thorsten Joachims. 2021. Fairness of exposure in stochastic bandits. In *International Conference on Machine Learning*. PMLR, 10686–10696.
[43] Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to rank with selection bias in personal search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 115–124.
[44] Hilde Weerts, Lambèr Royakkers, and Mykola Pechenizkiy. 2022. Does the End Justify the Means? On the Moral Justification of Fairness-Aware Machine Learning. *arXiv preprint arXiv:2202.08536* (2022).
[45] Tao Yang and Qingyao Ai. 2021. Maximizing marginal fairness for dynamic learning to rank. In *Proceedings of the Web Conference 2021*. 137–145.
[46] Xun Yang, Xiangnan He, Xiang Wang, Yunshan Ma, Fuli Feng, Meng Wang, and Tat-Seng Chua. 2019. Interpretable Fashion Matching with Rich Attributes. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 775–784. https://doi.org/10.1145/3331184.3331242
[47] Shoshana Zuboff. 2019. *The age of surveillance capitalism: The fight for a human future at the new frontier of power: Barack Obama's books of 2019*. Profile books. 121–125 pages.

Received May 10, 2023