Predictive Uncertainty-based Bias Mitigation in Ranking

Maria Heuss
University of Amsterdam, Amsterdam
The Netherlands
m.c.heuss@uva.nl

Daniel Cohen
Dataminr, NYC, USA
daniel.cohen@dataminr.com

Masoud Mansoury
University of Amsterdam
Discovery Lab, Elsevier
Amsterdam, The Netherlands
m.mansoury@uva.nl

Maarten de Rijke
University of Amsterdam, Amsterdam
The Netherlands
m.derijke@uva.nl

Carsten Eickhoff
University of Tübingen, Tübingen
Germany
c.eickhoff@acm.org

ABSTRACT

Societal biases that are contained in retrieved documents have received increased interest. Such biases, which are often prevalent in the training data and learned by the model, can cause societal harms, by misrepresenting certain groups, and by enforcing stereotypes. Mitigating such biases demands algorithms that balance the trade-off between maximized utility for the user with fairness objectives, which incentivize unbiased rankings. Prior work on bias mitigation often assumes that ranking scores, which correspond to the utility that a document holds for a user, can be accurately determined. In reality, there is always a degree of uncertainty in the estimate of expected document utility. This uncertainty can be approximated by viewing ranking models through a Bayesian perspective, where the standard deterministic score becomes a distribution.

In this work, we investigate whether uncertainty estimates can be used to decrease the amount of bias in the ranked results, while minimizing loss in measured utility. We introduce a simple method that uses the uncertainty of the ranking scores for an uncertainty-aware, post hoc approach to bias mitigation. We compare our proposed method with existing baselines for bias mitigation with respect to the utility-fairness trade-off, the controllability of methods, and computational costs. We show that an uncertainty-based approach can provide an intuitive and flexible trade-off that outperforms all baselines without additional training requirements, allowing for the post hoc use of this approach on top of arbitrary retrieval models.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**; **Evaluation of retrieval results**.

KEYWORDS

Mitigating bias, Fairness, Uncertainty, Utility-fairness trade-off

ACM Reference Format:
Maria Heuss, Daniel Cohen, Masoud Mansoury, Maarten de Rijke, and Carsten Eickhoff. 2023. Predictive Uncertainty-based Bias Mitigation in Ranking.

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10.
https://doi.org/10.1145/3583780.3615011

In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3583780.3615011

1 INTRODUCTION

The probability ranking principle (PRP) [36] states that, for optimal retrieval, the documents should be ranked in order of the predicted probability of relevance to the user. While this principle is ideal with respect to user utility, a ranking approach that solely relies on this principle can lead to an unfair treatment of the documents through unfair exposure and learned historical biases that are implicit in the data [3, 48]. This realization has led to a broad range of work in the field of *fair* ranking, where ways of ranking are explored that do not always strictly follow the PRP, but instead correct for such historical biases and distribute exposure more fairly [4, 13, 31]. Such biases can be reflected in different ways, e.g., models can be biased to over proportionally favor members of one group over another [1]. In this work, we follow Rekabsaz et al. [33], and say that a ranking model is *biased*, if documents that contain biases or stereotypes towards a protected group, e.g., people identifying with a certain gender, are being placed in ranked lists for queries that should be inherently neutral.

Using uncertainty to mitigate biases and improve fairness. Recent work has highlighted how learned ranking models violate the PRP – that each score is not well calibrated, and that learned ranking models do not provide an equally reliable estimate of a document's relevance [8, 32]. In this work, we take advantage of this violation of assumptions to produce a fair ranking with minimal utility loss. Rather than relying on a deterministic score, we consider the *uncertainty of the model's estimate* to violate the PRP in an informed manner by focusing on the most *uncertain documents*.

Our proposed method, called **P**redictive **U**ncertainty based **F**air **R**anking (PUFR) exploits knowledge about the certainty of the predicted relevance scores for mitigating bias by intervening at the scoring distribution, making it a post-processing method that is easy to use on top of arbitrary ranking models. Furthermore, PUFR does not require any training or fine-tuning of supervised models. Rather, given a ranked list of documents generated by a ranking model (most likely biased), PUFR leverages the uncertainty of the predicted scores assigned to the candidate documents by the ranking model to modify the ranked list among the most uncertain positions to generate a fairer ranking. PUFR aims to reduce the

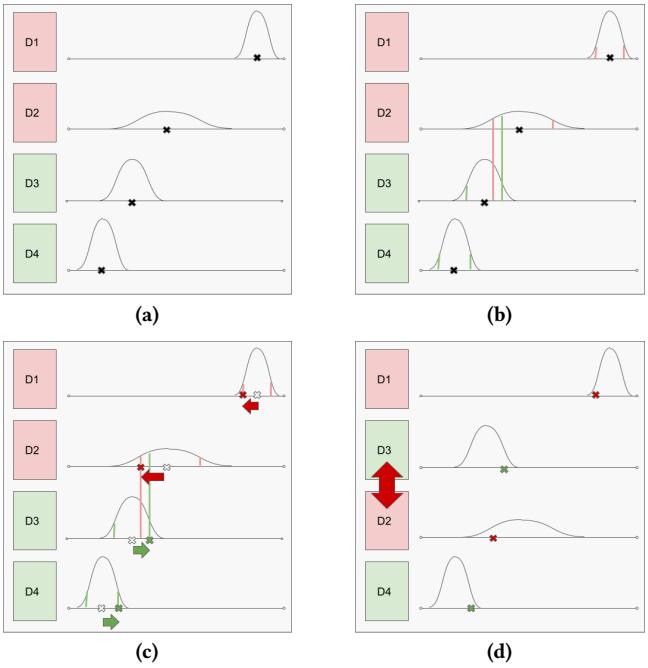


Figure 1: Visualization of our method PUFR. Next to the mean ranking scores PUFR also considers the score distribution that we obtained from the ranking model (1a). Through intersecting confidence intervals (1b) that allow us to adjust the scores (1c) such that a not biased document, visualized in green, is swapping place with a higher ranked, biased document (1d).

impact of biased documents, while adhering to the PRP as closely as possible, only intervening in places where the ranking model was not very certain to begin with.

Additionally, we introduce an entirely post hoc uncertainty quantification procedure, based on Laplace approximation, that allows PUFR to approximate the uncertainty for any off the shelf model without access to the training data or optimization procedure. This is in contrast to past work that requires a specific training regime to produce the uncertainty scores for each candidate [8, 9, 32, 47].

Motivating example. In Fig. 1, we visualize our approach to predictive uncertainty-based fairness, PUFR. In this example, the objective is to promote the unbiased documents (marked in green) to appear on top of the ranked result. We start by considering not only the mean ranking score but also the score distribution (uncertainty) as visualized with the cross resp. curve in Fig. 1a. We chose confidence intervals relative to the standard deviation in which we allow PUFR to adjust the scores for each document, as can be seen in Fig. 1b. Depending on whether a document is biased or not, we increase the score in this confidence interval if the document is unbiased or decrease it otherwise as visualized with the green/red crosses in Fig. 1c. As the confidence intervals of the second (D2) and third (D3) documents *intersect*, this changes the order of these scores. After re-ranking with respect to the newly obtained scores, the protected document D3 has swapped place with the non-protected document D2 as seen in Fig 1d. As there are minimal computational costs for PUFR, developers/users have the freedom to modify the trade-off between utility and fairness with minimal costs for their use-cases.

Our contributions. We summarize our contributions as follows:

- We introduce the notion of uncertainty-based fair ranking and analyze the potential of using the model uncertainty w.r.t. the ranking scores for bias mitigation.
- We define PUFR, an intuitive re-ranking approach that takes as input the ranking score distribution and calculates new ranking scores that can be used to create a less biased ranked list, while still preserving some certainty guarantees.
- We compare PUFR to several in- and post-processing bias mitigation methods and show that it outperforms all baselines, while being computationally much less expensive than some of them. Moreover, we demonstrate that PUFR is easily controllable with respect to the trade-off between fairness and utility, making it practical for use in real-life ranking applications.

2 RELATED WORK

2.1 Uncertainty in ranking

Zhu et al. [54] introduce the notion of considering a model's confidence when ranking documents. The authors view the confidence of a score based on the probabilistic model's own estimate – the variance. Alternatively, we can assume a Bayesian perspective that considers how well the training data support the current model. As this approach does not rely on a probabilistic ranking model, it complements current ranking regimes. Penha and Hauff [32] first introduce this notion of uncertainty into conversational retrieval by incorporating dropout into a BERT architecture at inference time. The ranking score is then modified by an uncertainty measure to improve the final re-ranking. Cohen et al. [9] suggest a similar approach for ad hoc retrieval where only the last layer's uncertainty is measured to offset both the complexity of a neural model and the size of the document set with similar re-ranking improvements. Yang et al. [46] extend the above work by leveraging the uncertainty estimate to improve the exploration of an online learning to rank model. Rather than performing uncertainty-aware re-ranking, the uncertainty estimate is used to take an optimistic perspective on candidate documents to reduce the exploitation bias commonly found in an online learning to rank setting.

2.2 Mitigating bias and fair ranking

Recent years have seen a broad range of research on uncovering and mitigating biases in different information retrieval systems, such as biases in talent pool [16] and resume search [7] and the reinforcement of gender biases through search engines [14]. Rekabsaz and Schedl [34] explore the extent to which documents with gender bias can be found in the retrieved results of different neural retrieval models. Other work focuses more on the mitigation of such biases [e.g., 33, 53], where models are optimized to contain fewer biased documents for queries that are inherently unbiased. Rekabsaz et al. [33] use adversarial learning to remove gender bias from the trained model, Zerveas et al. [53] optimize the query representation from a previously trained architecture instead.

Mitigating biases is often framed as a fairness task. Zehlike et al. [51, 52] introduce a classification framework for fair ranking approaches, which we partly use to position our work in the existing fair ranking literature. As opposed to score-based fairness [5, 21, 42, 45], where the ranking scores are assumed to be known,

in this work we focus on supervised learning to rank, where the ranking scores need to be determined with a ranking model.

A large body of work focuses on *merit-based* fairness, where the goal is to distribute the user attention in some way proportional to the merit of either individual documents (individual fairness [e.g., 19, 25, 38]) or groups of documents [e.g., 3, 39, 44]. In contrast, other work [e.g., 48, 50] focuses on *representational* fairness, which is concerned with removing historical biases from the ranking or representing documents from different groups fairly w.r.t. some demographic within the ranking.

Independently of the notion of fairness, we differentiate between pre-processing [24], in-processing [2, 3, 33, 39, 40, 48, 49, 53], and post-processing [11, 22, 50] approaches to fairness interventions. These methods come into play either before the model is being trained, adjust the model or training process itself, or intervene after the model has been trained and the ranking scores are determined.

PUFR is a *post-processing* approach that aims to mitigate bias (*representational* unfairness) as opposed to prior in-processing work on the same task [33, 53]. While other work on post-processing approaches [such as, e.g., 6, 48] intervene at the ranked output, our approach instead adjusts the score distribution. What distinguishes PUFR from prior work on fair ranking is that we aim to exploit the uncertainty that the ranking model has on the predicted relevance scores to increase the fairness of the rankings.

2.3 Uncertainty in fair ranking

Prior work at the intersection of uncertainty and fairness can be grouped into two categories. The first category deals with uncertainty introduced when group membership cannot be determined with confidence. Ghosh et al. [17] discover that, when group labels are inferred from data, the usage of fair ranking methods can invalidate fairness guarantees and even increase the disadvantage that protected groups might receive. Mehrotra and Vishnoi [28] follow up on this work and develop a fair ranking framework for cases where socially-salient group attributes cannot be determined with certainty but are assumed to follow a given probability distribution.

The other category, which contains, among others, our work, considers the predictive uncertainty stemming from imperfect prediction of merits and ranking scores. Yang et al. [47] are concerned with uncertainty in the relevance estimation. Unlike our work, the authors study an online setting where the relevance estimation is constantly updated. We target a static setting, not aiming to reduce the uncertainty for some exploration strategy but to exploit the uncertainty to obtain a better trade-off between fairness and utility.

Lastly, Singh et al. [41] are concerned with uncertainty in merit due to observations of secondary attributes instead of directly observing the merit. The authors suggest a probabilistic fairness framework in the presence of such uncertainty. Their work defines a notion of fairness that takes the uncertainty in the merit prediction into account, while we exploit uncertainty to, for example, correct for historic biases in the data and ranking model.

In summary, where existing methods either ignore the predictive uncertainty of ranking scores, aim to either reduce uncertainty, or take it into account when defining fairness, our work is the first to harness uncertainty to improve the fairness-utility trade-off.

3 METHOD

We take an uncertainty-based approach to post hoc bias mitigation in ranking. We exploit the model's uncertainty over the predicted ranking scores to manipulate the ranking in a way that benefits documents that do not contain biases, which results in a fairer ranked list. By staying within a certain confidence range, we minimize the potential cost to utility. Following prior work [28, 33], we frame the task as a fair ranking problem.

Our method operates entirely through principled machinery and allows us to trade-off between user utility and fairness by adjusting a single coefficient. Furthermore, an existing ranker can be used as-is, without the need to retrain it, making it possible to use and adjust it for various levels of fairness, with little additional costs.

Below, in Section 3.1, we start by defining our notation and the fair ranking task. In Section 3.2, we introduce our method PUFR that, assuming that the predictive uncertainty over the ranking scores is given, uses those uncertainty values to develop a fair ranking approach. Finally, in Section 3.3 we follow with a description of how to attain the uncertainty of a given deterministic ranking model over its scores at inference time.

3.1 Notation and preliminaries

Given a query q, we consider the task of ranking documents from a candidate set $\mathcal{D}_q = \{d_{q,i}\}_i$ w.r.t. their relevance, to q. Regarding measured user utility only, an ideal ranked list would be ordered by decreasing document relevance. We assume a ranking model has been trained to order the documents w.r.t. the relevance to the query by predicting relevance scores. Most rankers are deterministic, outputting only a single predicted relevance score, $\mu_{q,i}$. In Section 3.3 we will describe how to approximate the uncertainty of predicted scores for such a model. We write $\sigma_{q,i}$ for the standard deviation of the predicted score $\mu_{q,i}$ for document $d_{q,i}$. Note that we implicitly assume the score distribution to be Gaussian.

Prior work has shown that models that are trained solely for maximizing the measured utility can be biased and contain unfair representations of the resulting ranked lists [34]. In this work, as an additional objective, we aim to decrease the presence of biased documents in the ranked lists. We treat the task as a fair ranking problem, where we want to increase the exposure of the protected group $\mathcal{D}_q^P \subset \mathcal{D}_q$ of documents without biases and decrease the exposure of the non-protected group $\mathcal{D}_q^N \subset \mathcal{D}_q$ of documents that contain biases.

3.2 PUFR: Uncertainty-aware fairness

In this section, we introduce our post-processing fairness intervention method **P**redictive **U**ncertainty based **F**air **R**anking, PUFR. The core idea of PUFR is to take advantage of the uncertainty of the model over the predicted ranking scores to adjust these scores proportional to the standard deviation of the predictive distribution for each document, allowing fairness adjustments with minimal cost to the utility. For now, we treat the score distribution for each document, $\mathcal{N}(\mu_{q,i}, \sigma_{q,i}^2)$, as being given, but in Section 3.3 we describe how to obtain it for a deterministic ranker.

As the goal of PUFR is to mitigate bias and hence increase the fairness of the ranking system, PUFR accomplishes this by swapping some of the documents of the protected group, $\mathcal{D}_q^P$, with

higher ranked documents of the non-protected group, $\mathcal{D}_q^N$. Since the uncertainty of the scores for the documents within the same group can differ greatly, this allows for a tuned adjustment of the ranking scores where swaps only occur in settings where there exists a reasonable chance of the documents being equally relevant, quantified by the model's uncertainty, $\sigma_{q,i}$.

In other words, we allow PUFR to pick ranking scores that maximize fairness in intervals $[\mu_{q,i} - \alpha \cdot \sigma_{q,i}, \mu_{q,i} + \alpha \cdot \sigma_{q,i}]$, without re-ordering the documents within the same group. Here, α is a user defined hyper-parameter that quantifies the chance of a utility violation when performing this procedure. A higher value of α will result in a fairer ranking but at the cost of less accurate predicted scores, and hence potentially a drop in utility.

As shown in Algorithm 1, PUFR initially loops over all documents of the protected group $d_{q,i} \in \mathcal{D}_q^P$, sorted w.r.t. decreasing ranking score, $\mu_{q,i}$, see line 1. PUFR then increases the score as much as possible while staying within the confidence bounds, i.e.,

$$\tilde{\mu}_{q,i} = \mu_{q,i} + \alpha \cdot \sigma_{q,i}. \tag{1}$$

See line 2. To avoid intra-group swapping of documents, modified ranking scores are bounded by the lowest score of any higher ranked document within the same group:

$$\tilde{\mu}_{q,i} \leq \min_{\mathcal{D}_q^P, j \leq i}(\tilde{\mu}_{q,j}), \tag{2}$$

where j, i are rank positions, see line 3. Equivalently, for all documents of the non-protected group, $d_{q,i} \in \mathcal{D}_q^N$, we decrease the score as follows, this time starting with the document with the lowest ranking score (see line 5):

$$\tilde{\mu}_{q,i} = \mu_{q,i} - \alpha \cdot \sigma_{q,i}, \tag{3}$$

see line 6. Again, to avoid the same intra-group swapping for the non-protected group, we lower bound the adjusted scores by the maximum score of all documents in the same group that are ranked lower in the original ranking:

$$\tilde{\mu}_{q,i} \geq \max_{\mathcal{D}_q^N, j \geq i}(\tilde{\mu}_{q,j}). \tag{4}$$

See line 7. PUFR then uses these adjusted scores $\tilde{\mu}_{q,i}$ to re-rank the documents (line 9).

Note that even though we define PUFR for a setting with only one protected document group, it can be extended to several protected groups, that need to receive different treatments. Our approach allows us to adjust the strength of the score adjustment individually for each group, e.g., enabling a stronger correction for more disadvantaged groups, by allowing a group-wise choice of hyper-parameter α_g.

Many pre-trained ranking models do not output the uncertainty scores $\sigma_{q,i}$ that PUFR employs to reorder rankings. Thus we need a way to approximate the uncertainty scores $\sigma_{q,i}$ in a post-processing manner. Next, we show how to do this with the help of Laplace approximation.

3.3 Attaining uncertainty scores from a deterministic ranking model

The goal is to attain effective uncertainty scores, σ, from a ranking model at inference time; conventional uncertainty approaches fail to satisfy this condition [9, 32, 46, 47]. Past approaches have relied on a specific training regime – Monte Carlo (MC) dropout – to achieve an effective Bayesian model. As PUFR is a post hoc method, we leverage an alternative form of uncertainty, *Laplace approximation*, that can be applied to any already trained ranking model.

Algorithm 1 **P**redictive **U**ncertainty based **F**air **R**anking (PUFR)

Require: mean ranking scores $\{\mu_{q,i}\}_{d_{q,i} \in \mathcal{D}_q}$, standard deviation $\{\sigma_{q,i}\}_{d_{q,i} \in \mathcal{D}_q}$, control parameter α, groups $\mathcal{D}_q^P$, $\mathcal{D}_q^N$

1: **for all** $d_{q,i} \in \mathcal{D}_q^P$, sorted by decreasing $\mu_{q,i}$ **do**
2: $\quad \tilde{\mu}_{q,i} \leftarrow \mu_{q,i} + \alpha \cdot \sigma_{q,i}$
3: $\quad \tilde{\mu}_{q,i} \leftarrow \max_{\mathcal{D}_q^P, j \leq i}(\tilde{\mu}_{q,j})$
4: **end for**
5: **for all** $d_{q,i} \in \mathcal{D}_q^N$, sorted by increasing $\mu_{q,i}$ **do**
6: $\quad \tilde{\mu}_{q,i} \leftarrow \mu_{q,i} - \alpha \cdot \sigma_{q,i}$
7: $\quad \tilde{\mu}_{q,i} \leftarrow \min_{\mathcal{D}_q^N, j \geq i}(\tilde{\mu}_{q,j})$
8: **end for**
9: Obtain ranking L by sorting documents $d_{q,i} \in \mathcal{D}_q$ with respect to scores $\tilde{\mu}_{q,i}$
10: **return** L

The standard approach to training a deterministic model f, where there exists a single output for each input, is to learn a set of parameters, θ_{MAP}, that minimizes the loss function

$$\mathcal{L}(\theta) = -\ln P(\theta \mid \mathcal{D}) + r(\theta), \tag{5}$$

where r is some regularization on θ and $\mathcal{D}$ is the training dataset. While this is a probabilistic interpretation of the loss function and optimization process, prior work has mapped margin-based ranking losses to this framework [9]. At inference time, the model, f, is evaluated using the single point θ_{MAP}, which minimizes $\mathcal{L}(\theta)$. Alternatively, a Bayesian perspective captures the uncertainty of the model by considering all possible θ values weighed by how likely they are based on the training data using the posterior $P(\theta \mid \mathcal{D})$, with θ_{MAP} as the most likely value. This produces a distribution over outputs, of which the variance σ^2 represents the uncertainty present within the model and $\mathcal{D}$:

$$P(y \mid x, \mathcal{D}) = \int_\theta P(y \mid x, \theta) P(\theta \mid \mathcal{D}) d\theta, \tag{6}$$

with x as the input and y as the output of the model. Unfortunately, capturing this distribution is intractable for all but the smallest models due to the nature of computing the posterior $P(\theta \mid \mathcal{D})$. There exists prior work that approximates this distribution using MC Dropout [9, 32, 46, 47]. However, this requires a specific training regime, which would prevent the general application of PUFR to off-the-shelf architectures or previously trained ranking models.

Using Laplace approximation for post-hoc uncertainty approximation. We propose using Laplace approximations (LA), which can turn any conventionally trained deterministic model into a Bayesian model at inference time to produce the necessary σ values for PUFR [27]. LA encompass a family of approaches that fit a local Gaussian around the MAP estimate (5) via a second-order Taylor expansion of the log posterior:

$$\begin{aligned} \ln P(\theta \mid \mathcal{D}) \approx\ & \ln P(\theta_{\text{MAP}} \mid \theta) \\ & \frac{1}{2}(\theta - \theta_{\text{MAP}})^\intercal \bar{H} (\theta - \theta_{\text{MAP}}), \end{aligned} \tag{7}$$

where $\bar{H}$ is the expected Hessian at θ_{MAP}. The key observation is that the right side only requires the deterministic model, θ_{MAP}

Algorithm 2 Post hoc uncertainty estimation for single query

Require: pre-trained l-layer model f_θ, $\theta_{\text{MAP}} = [\theta^1_{\text{MAP}}, \ldots, \theta^l_{\text{MAP}}]$, query q, candidate documents $\mathcal{D}_q = \{d_{q,i}\}_i$, Monte Carlo sample size N.

1: **for all** $d_{q,i} \in \mathcal{D}_q$ **do**
2: $h^{l-1}_i, y = f_{\theta_{\text{MAP}}}(q, d_{q,i})$
3: $H \approx \text{diag}(F) = \text{diag}(\mathbb{E}\left[\nabla_{\theta^l} \ln P(y \mid q, d_{q,i}))^2\right]$
4: **for all** $j \in N$ **do**
5: $\{\theta\}^j_1 \sim \mathcal{N}(\theta^l{}_{\text{MAP}}, \text{diag}F^{-1})$
6: **end for**
7: $\mu_{q,i} = \frac{1}{N}\sum_{t=1}^{N} f_{\theta^l_t}(h^{l-1}_i)$
8: $\sigma^2_{q,i} = \frac{1}{N}\sum_{t=1}^{N} f_{\theta^l_t}(h^{l-1}_i)^2 - \left(\frac{1}{N}\sum_{t=1}^{N} f_{\theta^l_t}(h^{l-1}_i)\right)^2$
9: **end for**
10: **return** $\mu_{q,i}, \sigma_{q,i}\ \forall d_{q,i} \in \mathcal{D}_q$

to produce the log Bayesian posterior distribution on the left side. Then, to recover the full posterior, exponentiating both sides reveals the Gaussian functional form for θ,

$$\begin{aligned} P(\theta \mid \mathcal{D}) &\approx P(\theta_{\text{MAP}} \mid \mathcal{D}) - \\ &\quad \exp\left(\frac{1}{2}(\theta - \theta_{\text{MAP}})^\intercal \bar{H}(\theta - \theta_{\text{MAP}})\right) \\ &\approx \mathcal{N}(\theta_{\text{MAP}}, \bar{H}^{-1}). \end{aligned} \tag{8}$$

Thus, this approximation can take any twice differentiable off-the-shelf model and conveniently convert it to a Bayesian model at inference time by inverting the Hessian. While inverting to produce the covariance matrix is intractable for most models, we leverage past work by only inverting the last layers of a neural model to achieve actionable uncertainty estimates with near-zero cost [8, 9] (Algorithm 2, lines 2–3). While there exists a closed form linearization of Eq. 8, we are able to achieve sufficient efficiency using Monte Carlo sampling to capture the predictive distribution $P(y \mid x, f)$ by sampling from the Gaussian (line 5), $\mathcal{N}(\theta_{\text{MAP}}, \bar{H}^{-1})$ [10],

$$\begin{aligned} P(y \mid x, \mathcal{D}) &= \int_\theta P(y \mid x, \theta) P(\theta \mid \mathcal{D}) d\theta \\ &\approx \frac{1}{N}\sum_{t=1}^{N} p(y \mid x, \theta_t), \theta_t \sim \mathcal{N}(\theta_{\text{MAP}}, \bar{H}^{-1}). \end{aligned} \tag{9}$$

Furthermore, as the covariance matrix H^{-1} is viewed as independent to the training process, we do not need to use the original loss function either [23]. Lastly, for further efficiency, we exploit the property that the Hessian, H, is equivalent to the Fisher information matrix, F, at θ_{MAP}. As shown in Algorithm 2, we therefore approximate H by taking the diagonal of F, which is a common approximation regime (line 3) [18, 35].

After estimating $\mathcal{N}(\theta_{\text{MAP}}, \bar{H}^{-1})$ for the last layer of a neural model, we sample this distribution N times to produce N versions of the last layer, in order to produce $\mu_{q,\cdot}$ and $\sigma^2_{q,\cdot}$ as parameters of the predictive distribution $P(y \mid x, \mathcal{D}) = \mathcal{N}(\mu_{q,\cdot}, \sigma^2_{q,\cdot})$ (line 7–8). These parameters are then used by PUFR as described in Section 3.2 to debias the ranked list.

4 EXPERIMENTAL SETUP

We aim to answer the following research questions with our experiments: (RQ1) Based on empirical findings, are the uncertainty intervals around the ranking scores of a Bayesian ranking model sufficiently intersecting to allow for a re-ranking of documents, while staying within reasonable certainty bounds? (RQ2) Can PUFR be used to reduce the number of biased documents that are ranked on top of the list more effectively than prior methods? (RQ3) How do the various methods for fairness interventions compare with respect to controllability and computational efficiency?

There are four properties that we consider relevant to answer these questions: (i) We want to improve the fairness within the rankings. (ii) We want to do so with the least loss in utility possible. (iii) The next property is the controllability of the approach at hand. A human user/engineer should be able to easily adjust the trade-off between fairness and utility to fit their purposes. (iv) The last property is computational efficiency since this can also play a role when choosing a fairness method.

Next, we detail our experimental design. Then we discuss the evaluation metrics that we use to measure the four properties mentioned above (Section 4.2) and the dataset that we use (Section 4.3). Section 4.4 summarizes the baselines that we compare against.

4.1 Experimental design

We perform our experiments on a web search task, where for each query, the objective is to rank documents that might be relevant to that query. In addition to the requirement of being relevant to the user, the ranked list should not contain any gender biases for queries that are naturally non-gendered [33]. Therefore, we consider only non-gendered queries and expect a fair ranking model to not promote any documents that are biased towards some gender. See Section 4.3 for a discussion on the data used for this task.

To get an effective impression of the trade-off between utility and fairness, we perform a range of experiments per baseline, by varying some hyperparameter α. We define this hyperparameter individually for each baseline, based on the respective underlying algorithms (see Section 4.4).

To demonstrate the efficacy of PUFR on current search models, we use the BERT ranker introduced by Nogueira and Cho [30] as it represents a common language model architecture in current ranking regimes [12, 20, 26, 37]. Due to hardware constraints, we use Bert-Mini [43], a distilled four-layer version of BERT that performs comparably to the full model in search and other related tasks. We note that in the case of uncertainty modeling, Cohen et al. [9] demonstrate that a distilled model results in less expressive ranking uncertainty compared to larger variants of the same architecture on the same data. Thus, Bert-Mini represents a challenging setting and a conservative estimate of PUFR's performance.

To facilitate reproducibility of our work, all code and parameters are made available; see Section 7.

4.2 Evaluation

User utility and fairness are measured per query. To get a single score to compare across methods, we report the mean over all queries. We measure significance with paired t-tests, where we treat the results of each query as one sample.

User utility. To measure user utility, we use the nDCG metric (normalized discounted cumulative gain). We use different cut-offs to measure the user utility in the top-10 documents, as well as for the first 100 documents.

Fairness. As discussed in Section 4.1, our task entails reducing the impact of strongly biased documents in the presented rankings. Therefore, we use the nFaiRR metric as a measure of fairness introduced by Rekabsaz et al. [33]. For a ranked list L, the FaiRR score at cut-off k is defined as:

$$\text{FaiRR@k}(L) = \sum_{\text{rank}_L(d_i) \leq k} n_{d_i} \cdot \frac{1}{\text{rank}_L(d_i)}, \tag{10}$$

where $\text{rank}_L(d_i)$ describes the rank of candidate document d_i in L, and the neutrality score $n_{d_i} \in [0, 1]$ is lower, the more biased a document is. Since the possible range of FaiRR scores depends on the distribution of neutrality scores of its candidate documents, to make the results easier to interpret and better comparable among queries, we use the *normalized FaiRR score* (nFaiRR). For this, we normalize the FaiRR score with the highest attainable FaiRR score with the document candidates for this query, similar to how nDCG is calculated from DCG. In our experiments we measure the nFaiRR at a cut-off value of 10 and 50. We select a different cut-off than the utility measure (@100) so as to compare with reported values from the baseline evaluations.

Controllability. We follow prior work [33], and focus on a qualitative analysis of the results by investigating the predictability of the utility-fairness trade-off when adjusting the controllable hyperparameter of each of the methods. An ideal approach should have small change in utility and fairness for a small change in α. To this end, we compare the plots in Fig. 6 below.

Computational efficiency. For computational efficiency, we measure the run time of our implementation for each approach. We acknowledge that method-specific performance optimization might be able to further improve on the run times observed for the generic implementations used here, but assume that at least a rough execution time comparison can be gleaned. We measure the run time of each query and report the mean run time in Table 1.

Significance testing. To test the significance of observed differences in evaluation scores, we perform two-tailed paired t-tests on the metrics, treating the results of an approach of each query as a measurement of the same random variable. In Table 1, we mark results with an asterisk if they are significantly different from those of PUFR.

4.3 Dataset

The retrieval models that we use are trained on the MS MARCO Passage Retrieval collection [29]. For evaluation, we use MS $\text{MARCO}_{\text{Fair}}$, a subset consisting of 215 queries from the validation set that are non-gendered in nature – i.e., not containing any words or concepts that could be attributed to some gender [33]. However, the top candidate documents for these queries are highly associated with gender [33, 53]. We quantify the degree of gender bias for each document using the neutrality scores provided by Rekabsaz and Schedl [34] in order to measure fairness. We define documents with neutrality score 1 as the protected group for the post-processing baselines and PUFR.

4.4 Baselines

The baseline fairness intervention methods that we consider include the two in-processing approaches that have been introduced for the same bias mitigation task and dataset used here [33, 53]. Since PUFR is a post-processing approach, we add two commonly used post-processing fairness approaches that have been slightly adjusted to fit the task. Both post-processing baselines as well as UNFAIR use the mean scores $\mu_{q,i}$, produced by Algorithm 2 in Section 3.3 for the BERT-based ranker (see Section 4.1) as ranking scores. For each baseline the hyper-parameter α that allows us to control the trade-off between utility and fairness, is defined individually.

UNFAIR. The ranking resulting from ordering the documents with respect to the mean scores $\mu_{q,i}$, without considering fairness.

ADV. The (in-processing) adversarial fairness optimization from [33], which shares the same underlying BERT re-ranking architecture as discussed in Section 4.1. However, training is done using an adversarial discriminator head that attempts to predict whether the document is gendered or neutral by optimizing a classification loss function. The gradient from this loss is reversed within the main BERT architecture, therefore moving the parameters away from regions that can effectively capture gender [15]. We implement this model using the source code and suggested hyperparameters provided by the authors. The controlling hyperparameter α (originally λ) is defined by the scale of the reversed gradient.

CODER. This (in-processing) baseline [53] is intended for dense retrieval architectures. The method directly optimizes the query representation from a previously trained architecture, TAS-B [20], by jointly optimizing thousands of candidate documents in a list-wise manner. While improving overall ranking performance, the large candidate pool within a list-wise loss provides a stable and competitive way to incorporate fairness directly during training. We include this baseline not as a direct comparison with respect to ranking performance, but to provide context on how a direct list-based fairness optimization approach compares to methods that operate entirely within a post hoc framework when viewed from a utility-fairness trade-off perspective. Here, the hyperparameter α (in the original paper λ_r) is defined as the regularization coefficient for the neutrality loss.

CVXOPT. A (post-processing) convex optimization approach similar to [6]. For each query we optimize the ranking L for utility, measured by nDCG, under a constraint on the nFaiRR score, $\text{nFaiRR}(L) \geq \alpha$. To keep computational costs within a reasonable range, we only re-rank the first 50 documents of each query.

FA*IR. A (post-processing) approach suggested in [48]. We use a significance parameter 0.1 as suggested in [48] and vary p, the desired minimal proportions of documents with the protected attribute in the top-k for any value of k. In the remainder of this paper we use $\alpha := p$, not to be confused with the significance parameter in the original paper, to match the other methods. For a fair comparison w.r.t. to computational efficiency, we use an efficient implementation that pre-computes the required number of protected documents for each rank upfront via an iterative algorithm.

5 EXPERIMENTAL RESULTS

We present and discuss answers to our research questions.

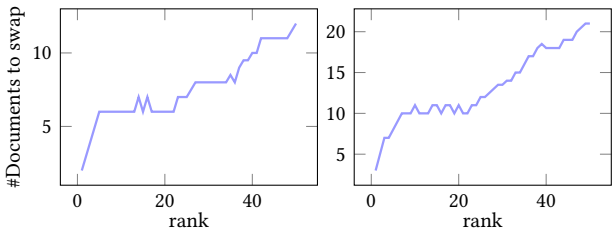


Figure 2: MSMARCO$_{\text{FAIR}}$: Median number of documents that have intersecting uncertainty intervals with the document placed at each rank for uncertainty intervals of 1 (left) resp. 2 (right) standard deviations.

5.1 Intersections of uncertainty intervals

Recall (RQ1): *Based on empirical findings, are the uncertainty intervals around the ranking scores of a Bayesian ranking model sufficiently intersecting to allow for a re-ranking of documents, while staying within reasonable certainty bounds?* To answer (RQ1), we analyze the confidence intervals of the ranking scores. If the uncertainty intervals do not intersect much, the ranking model is very certain about the ordering of its ranking scores. In such a case, our approach, or any uncertainty-aware approach in general, would not be able to re-rank the documents within an acceptable utility bracket. Previous work has shown that ranking models tend to be very certain for the ranking scores of highly ranked documents [9], but certainty decays when going down the ranked list. We are interested in how much flexibility a rank-aware fairness approach would offer in swapping documents by allowing the ranking scores to take values in a given certainty $[\mu_{q,i} - \alpha \cdot \sigma_{q,i}, \mu_{q,i} + \alpha \cdot \sigma_{q,i}]$ interval around the mean score value $\mu_{q,i}$. Fig. 2 shows the median number of documents with intersecting confidence intervals (i.e. the median number of documents that the document at that rank could swap position with) for $\alpha = 1$ resp. $\alpha = 2$ standard deviations.

Even for documents ranked at higher positions, there is flexibility to change the order of the ranking. For a confidence interval of 1 standard deviation, most documents in the top-10 each have at least 6 documents that they could swap rank with. If we look at confidence intervals of two standard deviations, this number increases to ~10 documents that the document at rank 10 can swap place with. We therefore answer (RQ1) positively: The uncertainty intervals around the ranking scores of the Bayesian ranking model are sufficiently intersecting to allow for a re-ranking of documents, while staying within acceptable certainty bounds for utility.

Having confirmed that within the uncertainty of the model there is flexibility for an uncertainty-based fairness approach to change the order of documents, we address our second research question that asks whether the proposed approach can improve fairness.

5.2 The fairness utility trade-off

Recall (RQ2): *Can PUFR be used to reduce the number of biased documents that are ranked on top of the list more effectively than prior methods?* To answer this question we refer to Fig. 3 and 4, where we plot fairness on the x-axis against utility on the y-axis, for PUFR and the baselines discussed in Section 4.4, for different values of the respective hyper-parameter α that controls the trade-off. In addition we use Table 1, where we compare the experimental

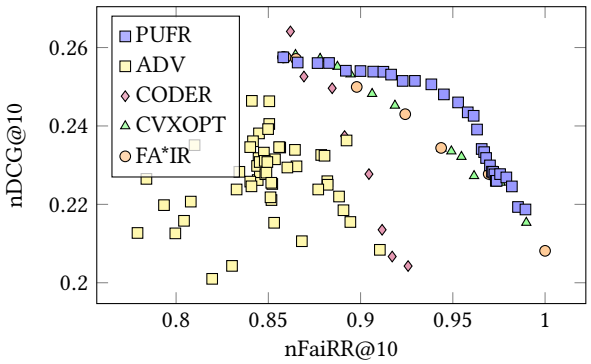


Figure 3: Trade-off between fairness and utility evaluated on the first 10 documents.

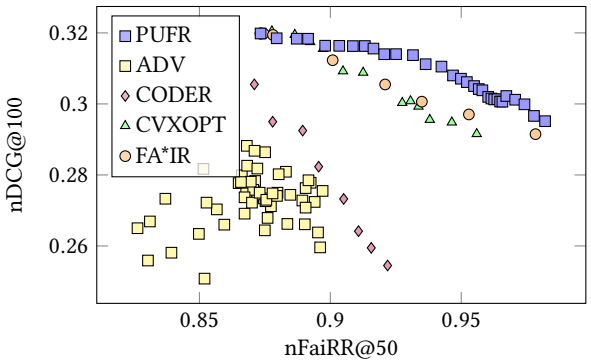


Figure 4: Trade-off between fairness and utility evaluated on the first 50 resp. 100 documents.

outcomes with the best nFaiRR value for a given minimum utility requirement.

Utility-fairness trade-off. In Fig. 3 and 4, we observe that the CODER baseline starts with a better trade-off for the top-10 documents, which can be attributed to better ranking scores that it starts out with (PUFR uses a BERT-based model to obtain ranking scores). CODER's advantage quickly vanishes as the balancing parameter α increases for more weight on fairness. Overall, PUFR offers a better trade-off between fairness and utility than the CODER based and the adversarial fairness optimization baseline (ADV).

If we compare PUFR to the post-processing baselines (CVXOPT and FA*IR), it clearly outperforms those baselines. Once a nFaiRR value of 0.96 is reached the advantage of PUFR over these baselines becomes smaller. For a possible explanation see Section 6.

Overall, PUFR outperforms all baselines for a large range of nFaiRR values, which we also highlight by comparing the fairness of the different approaches at two different utility levels (nDCG@100 = 0.31 and nDCG@100 = 0.30) in Table 1. We chose these levels of utility, assuming that, when taking a fair ranking approach in production there might be a certain (small) allowance for a drop in utility given, within which the best possible fairness value should be reached. We see that for these levels PUFR reaches significantly higher scores for nFaiRR than all baselines.

Ablation study. To ensure that the uncertainty estimates indeed do contribute to the success of PUFR, we conduct an ablation study. We compare PUFR with a similar approach that, instead of adjusting the scores relative to the standard deviation, in- or decreases all

Table 1: Results for experiment with best nFairr value for nDCG decrease not more than 0.01 and 0.02 respectively. ADV baseline does not fulfill the criteria of being at most 0.01 nDCG points worse than UNFAIR. * denotes significance w.r.t. PUFR via two tailed paired students t-test of $p < .05$.

			nDCG↑		nFaiRR↑		re-rank-	req.
	Method	α	@10	@100	@10	@50	time(s)↓	train
	UNFAIR	0.0	0.26	0.32	0.858	0.873	0.00	No
	ADV	2.0	0.21	0.26	0.91	0.896	-	Yes
$nDCG_{100} \geq 0.31$	**PUFR**	2.5	**0.25**	**0.31**	**0.938**	**0.932**	0.014	No
	CODER	3.0	**0.25**	**0.31**	0.920*	0.920*	-	Yes
	CVXOPT	0.8	**0.25**	**0.31**	0.906*	0.905*	0.123	No
	FA*IR	0.7	**0.25**	**0.31**	0.898*	0.901*	0.058	No
$nDCG_{100} \geq 0.30$	**PUFR**	7.0	0.23	**0.30**	**0.970**	**0.960**	0.014	No
	CODER	4.0	**0.24**	**0.30**	0.927*	0.926*	-	Yes
	CVXOPT	0.91	0.23	**0.30**	0.949*	0.931*	0.123	No
	FA*IR	0.85	0.23	**0.30**	0.944*	0.935*	0.058	No

scores by the same, constant value. In our experiments we use the mean uncertainty score over all queries and candidates documents, $\sigma_{\text{mean}} = \text{mean}_{q,i}(\sigma_{q,i})$. The results of this ablation study are presented in Fig. 5. We see that by using the uncertainty scores instead of a uniform correction factor, we gain a better trade-off. For the top-10, these improvements are less visible (see Fig. 5 (a)). When considering the top-100 documents instead, the advantages of using uncertainty become much clearer (see Fig. 5 (b)). This might be due to fact that, as also noted by Cohen et al. [9], for the top-10 documents the uncertainty scores tend to be fairly similar to each other, making our approach, if we only look at a small window, seem similar to the ablation study approach. When we look at a larger window, the uncertainty scores deviate more, emphasizing the advantages of PUFR.

We conclude this section and answer (RQ2) in the affirmative. PUFR performs competitively with baselines. In terms of fairness-utility trade-offs it significantly outperforms other post-processing schemes, and clearly beats the two state-of-the-art in-processing baselines. The ablation study confirms that this result is at least partially due to the use of the model's uncertainty in its scores. Hence, PUFR can be used to reduce the number of biased documents that are ranked on top of the list more effectively than prior methods.

Since a good utility-fairness trade-off is not the only relevant criterion when choosing a fair ranking method, our next research question (RQ3) concerns the degree of controllability and computational costs of the different methods.

5.3 Controllability and computational efficiency

Next, we address (RQ3): *How do the various methods compare with respect to controllability and computational efficiency?* As discussed in Section 4.2, we focus on a qualitative analysis of the α-fairness and α-utility curves, evaluating how predictable and hence controllable the utility-fairness trade-off is. Fig. 6 shows that for PUFR the nFaiRR score monotonically increases with increasing α. At the same time, utility, measured by nDCG, decreases. Both curves are highly predictable. Furthermore, since re-ranking is computationally very efficient, a broad range of rankings with different trade-offs

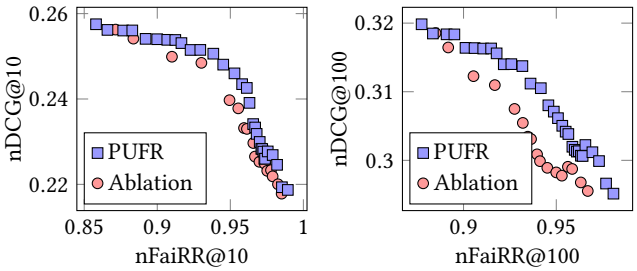


Figure 5: Ablation study comparing PUFR (score adjustment proportional to the ranker's uncertainty) with an ablation experiment with uniform score adjustment.

can be explored to find the right choice of hyper-parameter for the desired trade-off between nFaiRR and nDCG. The CODER-based approach has similarly predictable trade-off curves as PUFR [53]. However, CODER is an in-processing approach, meaning that the model needs to be re-trained for each choice of hyper-parameter α, making it much less controllable in practice. The ADV method on the other hand, seems to be highly unpredictable, on top of the downsides that come with in-processing methods as discussed above. For the FA*IR baseline, although its curve seems to be fairly well controllable, the granularity in which we can produce results is much coarser. Due to space constraints we omit the figure for the convex optimization approach; because of computational efficiency, FA*IR or PUFR should be preferred over it.

With regard to computational efficiency, we recall that both in-processing approaches, ADV and CODER, once trained, do not have the post-processing overhead of the other methods. However, these methods need a large amount of training to gain a reasonable level of performance [33, 53]. Looking at Table 1, re-ranking with PUFR is much faster than with the other two post-processing approaches. Obtaining uncertainty labels can be done within microseconds. After adjusting the ranking scores there is a single re-sorting of the documents that dominates the execution time. Hence, when using PUFR in production and adjusting the score before the initial ordering of the documents, the execution of PUFR is nearly free.

6 DISCUSSION

Exploiting model uncertainty for the fairness-utility trade-off. To increase the fairness of a ranking, we would commonly need to trade-off some predicted utility.Encouraging this trade-off to take place when the ranking model is less certain about the ranking scores will cause roughly equivalently relevant documents that the model cannot confidently rank, to swap place. Assuming that the ranking model is well calibrated, this might be the reason for the overall better trade-off that PUFR achieves, compared to models that do not consider predictive uncertainty. This quality is highlighted in Fig. 7, where we show the score distribution of the top-5 documents of two queries in the $\text{MSMARCO}_{\text{Fair}}$ dataset. In the case of Fig. 7a and 7b, the larger variance leads to overlapping score distribution, allowing PUFR to swap documents in the re-ranked list. On the other hand, Fig. 7c and 7d show a query where the model is very certain about the order of the documents. PUFR hence does not change the order of the documents, whereas FA*IR and CVXOPT both do adjust the ranking, leading to decreased user utility for those baselines.

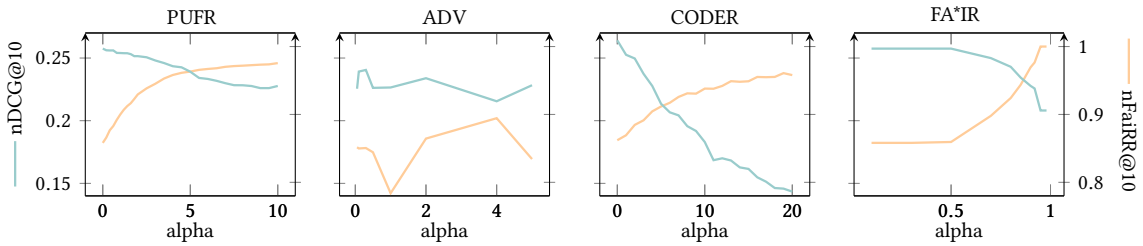
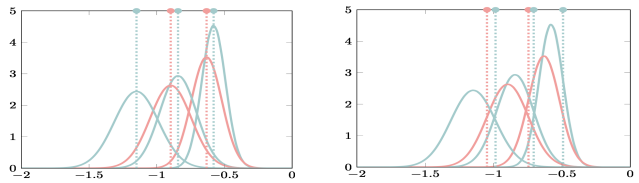
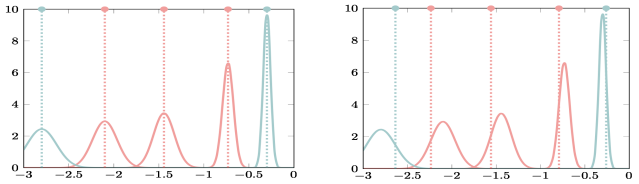


Figure 6: Controllability of different approaches visualized by plotting utility and fairness against the controlling hyper-parameter α on the x-axis (see Section 4.4 for a description of α for each approach).



(a) query 1032855 before PUFR **(b) query 1032855 after PUFR**



(c) query 1089383 before PUFR **(d) query 1089383 after PUFR**

Figure 7: Examples of score distributions for the top-5 documents for two queries of the MS MARCO$_{\text{Fair}}$ dataset. Protected documents in green, non-protected in red. Subfigs. 7a and 7c show the ranking score before PUFR adjusts the scores, 7b and 7d show them after. Query 1089383 was scaled before plotting.

Using PUFR outside the models confidence. Our empirical results show that if we allow PUFR to adjust the scores too far outside of its confidence, its performance starts to decay (see Fig. 3). If α is too high, the natural interpretation of adjusting the scores within plausible error-bounds gets lost and we cannot exploit the models knowledge of its own certainty any further. Without the certainty to back it up, PUFR becomes more arbitrary in its decisions where to trade-off predicted utility with fairness. Hence, PUFR is most effective for small values of α, roughly up to $\alpha = 4$ (see Fig. 6). This observation means that a purely uncertainty-based fairness method might not be the best choice when the bias we want to correct for is too strong. In such cases, it might be beneficial to use uncertainty in combination with another approach that has proven effective for the task at hand.

7 CONCLUSION

We have introduced the notion of predictive uncertainty-based ranking fairness, aiming to exploit a ranking model's uncertainty as an indicator of which documents we should focus on when re-ordering for a fairer ranking which de-emphasizes documents containing biases. Through our empirical analysis we have found that the uncertainty intervals of the ranking scores are sufficiently intersecting to allow us to swap the position of some documents. We have also introduced an intuitive and principled post-processing method, PUFR, that adjusts the predicted ranking scores within some desired confidence bound. We have shown that by considering uncertainty, PUFR can achieve the best utility-fairness trade-off and has superior time complexity and good controllability.

We hope that our contribution makes the adoption of methods to remove bias in ranked results more attractive to practitioners working on real- world search and recommendation systems.

More experimentation is needed to confirm our findings in more settings. We see limitations of our approach as twofold. Firstly, PUFR allows a re-ordering of the documents only within the uncertainty of the model. This might make our method less effective in reducing unfairness when the model is very skewed towards documents containing biases. As a second limitation, we rely on uncertainty scores containing accurate information on which documents are more likely to be in the wrong order. Furthermore, the uncertainty intervals around the scores need to intersect sufficiently. In our experiments, we are using a neural ranking model on text data, which is a task that inherently carries a fair amount of uncertainty. For other tasks and fairness definitions, more research will be necessary to evaluate whether an uncertainty-based approach can be beneficial for the utility-fairness trade-off.

As to future work, an important next step would be to define ways to evaluate uncertainty scores in a listwise manner for ranking models. Without proper evaluation of the predictive uncertainty, we are unable to put trust on the score distribution and hence on an uncertainty-based fairness approach. Moreover, more work is needed to investigate whether PUFR could be extended to, for example, Bayesian learning-to-rank models or recommender systems. Finally, we see a clear need to create more datasets for large language models with fairness labels, on which methods such as ours can be tested.

Data and code. To facilitate reproducibility of our work, all code and parameters are shared at https://github.com/MariaHeuss/2023-CIKM-uncertainty-based-bias-mitigation.

ACKNOWLEDGMENTS

The research was partially funded by the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl, and project LESSEN with project number NWA.1389.20.183 of the research program NWA ORC 2020/21, which is (partly) financed by the Dutch Research Council (NWO). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine Bias. In *Ethics of Data and Analytics*. Auerbach Publications, 254–264.

[2] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H. Chi, and Cristos Goodrow. 2019. Fairness in Recommendation Ranking through Pairwise Comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2212–2220.

[3] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of Attention: Amortizing Individual Fairness in Rankings. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 405–414.

[4] Carlos Castillo. 2019. Fairness and Transparency in Ranking. In *ACM SIGIR Forum*, Vol. 52. 64–71.

[5] L. Elisa Celis, Anay Mehrotra, and Nisheeth K. Vishnoi. 2020. Interventions for Ranking in the Presence of Implicit bias. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 369–380.

[6] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. 2018. Ranking with Fairness Constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. 28:1–28:15.

[7] Le Chen, Ruijun Ma, Anikó Hannák, and Christo Wilson. 2018. Investigating the Impact of Gender on Rank in Resume Search Engines. ACM, 1–14.

[8] Daniel Cohen, Kevin Du, Bhaskar Mitra, Laura Mercurio, Navid Rekabsaz, and Carsten Eickhoff. 2022. Inconsistent Ranking Assumptions in Medical Search and Their Downstream Consequences. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain). ACM, 2572–2577.

[9] Daniel Cohen, Bhaskar Mitra, Oleg Lesota, Navid Rekabsaz, and Carsten Eickhoff. 2021. Not All Relevance Scores Are Equal: Efficient Uncertainty and Calibration Modeling for Deep Retrieval Models. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 654–664.

[10] Zhijie Deng, Feng Zhou, and Jun Zhu. 2022. Accelerated Linearized Laplace Approximation for Bayesian Deep Learning. In *Advances in Neural Information Processing Systems*, Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.), Vol. 35. 2695–2708.

[11] 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.

[12] Qian Dong, Yiding Liu, Suqi Cheng, Shuaiqiang Wang, Zhicong Cheng, Shuzi Niu, and Dawei Yin. 2022. Incorporating Explicit Knowledge in Pre-Trained Language Models for Passage Re-Ranking. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1490–1501.

[13] Michael D. Ekstrand, Graham McDonald, Amifa Raj, and Isaac Johnson. 2022. Overview of the TREC 2021 Fair Ranking Track. In *The Thirtieth Text REtrieval Conference (TREC 2021) Proceedings*.

[14] Alessandro Fabris, Alberto Purpura, Gianmaria Silvello, and Gian Antonio Susto. 2020. Gender Stereotype Reinforcement: Measuring the Gender Bias Conveyed by Ranking Algorithms. *Information Processing & Management* 57 (2020), 102377.

[15] Yaroslav Ganin and Victor S. Lempitsky. 2015. Unsupervised Domain Adaptation by Backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015 (JMLR Workshop and Conference Proceedings, Vol. 37)*. 1180–1189.

[16] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. 2019. Fairness-aware Ranking in Search & Recommendation Systems with Application to Linkedin Talent Search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2221–2231.

[17] Avijit Ghosh, Ritam Dutt, and Christo Wilson. 2021. When Fair Ranking Meets Uncertain Inference. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1033–1043.

[18] Sruthi Gorantla, Amit Deshpande, and Anand Louis. 2021. On the Problem of Underranking in Group-Fair Ranking. In *International Conference on Machine Learning*. PMLR, 3777–3787.

[19] Maria Heuss, Fatemeh Sarvi, and Maarten de Rijke. 2022. Fairness of Exposure in Light of Incomplete Exposure Estimation. In *SIGIR 2022: 45th international ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 759–769.

[20] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 113–122.

[21] Jon Kleinberg and Manish Raghavan. 2018. Selection Problems in the Presence of Implicit Bias. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 33:1–33:17.

[22] Till Kletti, Jean-Michel Renders, and Patrick Loiseau. 2022. Pareto-Optimal Fairness-Utility Amortizations in Rankings with a DBN Exposure Model. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 748–758.

[23] Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. 2020. Being Bayesian, Even Just a Bit, Fixes Overconfidence in ReLU Networks. In *Proceedings of the 37th International Conference on Machine Learning*. 5392–5402.

[24] Preethi Lahoti, Krishna P. Gummadi, and Gerhard Weikum. 2019. iFair: Learning Individually Fair Data Representations for Algorithmic Decision Making. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 1334–1345.

[25] Preethi Lahoti, Krishna P. Gummadi, and Gerhard Weikum. 2019. Operationalizing Individual Fairness with Pairwise Fair Representations. *Proceedings of the VLDB Endowment* (2019), 506–518.

[26] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2021. Pretrained Transformers for Text Ranking: BERT and Beyond. *Synthesis Lectures on Human Language Technologies* 14, 4 (2021), 1–325.

[27] David J. C. Mackay. 1992. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation* 4 (1992), 448–472.

[28] Anay Mehrotra and Nisheeth Vishnoi. 2022. Fair Ranking with Noisy Protected Attributes. In *Advances in Neural Information Processing Systems*, Vol. 35. 31711–31725.

[29] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated Machine Reading Comprehension Dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches 2016*.

[30] Rodrigo Frassetto Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085* (2019).

[31] Gourab K. Patro, Lorenzo Porcaro, Laura Mitchell, Qiuyue Zhang, Meike Zehlike, and Nikhil Garg. 2022. Fair Ranking: A Critical Review, Challenges, and Future Directions. In *2022 ACM Conference on Fairness, Accountability, and Transparency*. ACM, 1929–1942.

[32] Gustavo Penha and Claudia Hauff. 2021. On the Calibration and Uncertainty of Neural Learning to Rank Models for Conversational Search. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*. Association for Computational Linguistics, 160–170.

[33] Navid Rekabsaz, Simone Kopeinik, and Markus Schedl. 2021. Societal Biases in Retrieved Contents: Measurement Framework and Adversarial Mitigation of Bert Rankers. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 306–316.

[34] Navid Rekabsaz and Markus Schedl. 2020. Do Neural Ranking Models Intensify Gender Bias?. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2065–2068.

[35] Hippolyt Ritter, Aleksandar Botev, and David Barber. 2018. A Scalable Laplace Approximation for Neural Networks. In *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*. International Conference on Representation Learning.

[36] Stephen E Robertson. 1977. The Probability Ranking Principle in IR. *Journal of Documentation* 33 (1977), 294–304.

[37] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. ACM, 3715–3734.

[38] Fatemeh Sarvi, Maria Heuss, Mohammad Aliannejadi, Sebastian Schelter, and Maarten de Rijke. 2022. Understanding and Mitigating the Effect of Outliers in Fair Ranking. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 861–869.

[39] 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.

[40] Ashudeep Singh and Thorsten Joachims. 2019. Policy Learning for Fairness in Ranking. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. 5426–5436.

[41] Ashudeep Singh, David Kempe, and Thorsten Joachims. 2021. Fairness in Ranking under Uncertainty. In *Advances in Neural Information Processing Systems*, Vol. 34. 11896–11908.

[42] Julia Stoyanovich, Ke Yang, and HV Jagadish. 2018. Online Set Selection with Fairness and Diversity Constraints. In *21st International Conference on Extending Database Technology, EDBT 2018*. 241–252.

[43] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-Read Students Learn Better: The Impact of Student Initialization on Knowledge Distillation. *arXiv preprint arXiv:1908.08962* (2019).

[44] Lequn Wang and Thorsten Joachims. 2021. User Fairness, Item Fairness, and Diversity for Rankings in Two-Sided Markets. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*. 23–41.

[45] Ke Yang, Vasilis Gkatzelis, and Julia Stoyanovich. 2019. Balanced Ranking with Diversity Constraints. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI*.

[46] Tao Yang, Chen Luo, Hanqing Lu, Parth Gupta, Bing Yin, and Qingyao Ai. 2022. Can Clicks Be Both Labels and Features? Unbiased Behavior Feature Collection

and Uncertainty-Aware Learning to Rank. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 6–17.

[47] Tao Yang, Zhichao Xu, Zhenduo Wang, Anh Tran, and Qingyao Ai. 2023. Marginal-Certainty-aware Fair Ranking Algorithm. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 24–32.

[48] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. FA*IR: A Fair Top-k Ranking Algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1569–1578.

[49] Meike Zehlike and Carlos Castillo. 2020. Reducing Disparate Exposure in Ranking: A Learning To Rank Approach. In *Proceedings of The Web Conference 2020*. 2849–2855.

[50] Meike Zehlike, Tom Sühr, Ricardo Baeza-Yates, Francesco Bonchi, Carlos Castillo, and Sara Hajian. 2022. Fair Top-k Ranking with Multiple Protected Groups. *Information Processing & Management* 59, 1 (2022), 102707.

[51] Meike Zehlike, Ke Yang, and Julia Stoyanovich. 2022. Fairness in Ranking, Part I: Score-based Ranking. *Comput. Surveys* 55, 6 (2022), 1–36.

[52] Meike Zehlike, Ke Yang, and Julia Stoyanovich. 2022. Fairness in Ranking, Part II: Learning-to-rank and Recommender Systems. *Comput. Surveys* 55, 6 (2022), 1–41.

[53] George Zerveas, Navid Rekabsaz, Daniel Cohen, and Carsten Eickhoff. 2022. Mitigating Bias in Search Results Through Contextual Document Reranking and Neutrality Regularization. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2532–2538.

[54] Jianhan Zhu, Jun Wang, Michael Taylor, and Ingemar J. Cox. 2009. Risk-Aware Information Retrieval. In *Advances in Information Retrieval: 31th European Conference on IR Research, ECIR 2009, Toulouse, France, April 6-9, 2009. Proceedings 31*. Springer-Verlag, 17–28.