Quantifying Query Fairness Under Unawareness

Thomas Jaenich*
t.jaenich.1@research.gla.ac.uk
University of Glasgow
Glasgow, UK

Alejandro Moreo*
alejandro.moreo@isti.cnr.it
Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche
56124 Pisa, Italy

Alessandro Fabris
alessandro.fabris@mpi-sp.org
Max Planck Institute for Security and Privacy (MPI-SP)
Bochum, Germany

Graham McDonald
Graham.Mcdonald@glasgow.ac.uk
University of Glasgow
Glasgow, UK

Andrea Esuli
andrea.esuli@isti.cnr.it
Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche
56124 Pisa, Italy

Iadh Ounis
Iadh.Ounis@glasgow.ac.uk
University of Glasgow
Glasgow, UK

Fabrizio Sebastiani
fabrizio.sebastiani@isti.cnr.it
Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche
56124 Pisa, Italy

Abstract

Traditional ranking algorithms are designed to retrieve the most relevant items for a user's query, but they often inherit biases from data that can unfairly disadvantage vulnerable groups. Fairness in information access systems (IAS) is typically assessed by comparing the distribution of groups in a ranking to a target distribution, such as the overall group distribution in the dataset. These fairness metrics depend on knowing the true group labels for each item. However, when groups are defined by demographic or sensitive attributes, these labels are often unknown, leading to a setting known as "fairness under unawareness." To address this, group membership can be inferred using machine-learned classifiers, and group prevalence is estimated by counting the predicted labels. Unfortunately, such an estimation is known to be unreliable under dataset shift, compromising the accuracy of fairness evaluations. In this paper, we introduce a robust fairness estimator based on quantification that effectively handles multiple sensitive attributes beyond binary classifications. Our method outperforms existing baselines across various sensitive attributes and, to the best of our knowledge, is the first to establish a reliable protocol for measuring fairness under unawareness across multiple queries and groups.

1. Introduction

In addition to ensuring the relevance of search results, preventing unfairness and discrimination in ranking has become a fundamental objective in the development of information access systems (IAS) (Ekstrand, Das, Burke, Diaz, et al., 2022a; Zehlike, Yang, & Stoyanovich, 2022). With this in mind, there have been many approaches proposed in the literature for mitigating unfairness in the results of IAS (Biega, Gummadi, & Weikum, 2018; Geyik, Ambler, & Kenthapadi, 2019; Heuss, Sarvi, & de Rijke, 2022; Jaenich, McDonald, & Ounis, 2023, 2024; Morik, Singh, Hong, & Joachims, 2020; Singh & Joachims, 2018). Providing fair search results is crucial, since ranking items that belong to groups identified by sensitive attributes can significantly impact real-world outcomes, such as economic opportunities (Chen, Ma, Hannák, & Wilson, 2018; Pedreschi, Ruggieri, & Turini, 2008). When items that are associated with a specific demographic attribute are systematically ranked lower in the search results than items from other demographics, the low-ranked items will receive less attention from users, since items high up in the ranking are more likely to be examined by users (Craswell, Zoeter, Taylor, & Ramsey, 2008). This can be problematic in practical scenarios. For example, in job search, this positional bias means that recruiters may only notice candidates from the top-ranked applications, potentially overlooking qualified individuals who are ranked lower. While this will not always warrant a bias-mitigating intervention, it is imperative to at least *evaluate* and *monitor* the fairness of rankings in high-stakes domain (European Commission, 2024; New York City Council, 2021).

Another important factor influencing the fairness of search results is the query that is issued by the user. The degree of unfairness in the search results can vary across different queries, depending on how a query is formulated. Such unfairness can be introduced either directly by the user, e.g., through the replication of existing biases when formulating the query (Kopeinik, Mara, Ratz, Krieg, Schedl, & Rekabsaz, 2023), or automatically, e.g., through the auto-completion features of a search system (Chen et al., 2018). Therefore, assessing the fairness of search results related to a specific query is crucial to determine whether fairness interventions are needed. We introduce the terminology *query fairness estimation* (QFE) for the task of assessing the fairness of search results for a given query.

To perform QFE, one typically needs access to the group labels of the ranked items (Kuhlman, Gerych, & Rundensteiner, 2021; Raj & Ekstrand, 2022; Zehlike et al., 2022). These labels categorise items by sensitive attributes, such as race or gender. However, access to these group labels is often limited due to legal, ethical, or other data availability constraints (Bogen, Rieke, & Ahmed, 2020a; Holstein, Wortman Vaughan, Daumé III, Dudik, & Wallach, 2019). As a result, fairness evaluations must often occur under "unawareness," where the labels are unknown.

One way to obtain the labels under unawareness is to use human annotators. However, this is costly and impractical in most scenarios. An alternative is to deploy classifiers to infer the document labels automatically. A classification method deployed in practice is the *Bayesian improved surname geocoding* (BISG) that is used to infer race from surnames and ZIP codes using Bayesian statistics and US Census data (Adjaye-Gbewonyo, Bednarczyk, Davis, & Omer, 2014). While cost-efficient, deploying classifiers can introduce more unintended unfairness and bias by the classifiers themselves (Ghosh, Dutt, & Wilson, 2021).

Moreover, even seemingly accurate classifiers can lead to unreliable results when performing QFE. For example, a good classifier may achieve high accuracy by focusing disproportionately on one class, thus minimising errors like false positives at the expense of increasing errors like false negatives (see §1.2 in Esuli, Fabris, Moreo, & Sebastiani, 2023). While technically accurate, this skewed performance fails to reflect the true distribution of groups, making it unreliable for assessing the proportions in a broader set of documents, and can lead to significant errors and misjudgments in the measurements of fairness (Chen, Kallus, Mao, Svacha, & Udell, 2019).

In this work, we propose the use of quantification techniques, i.e., machine learning models specifically trained to estimate the relative frequencies of the classes in unlabelled data (Esuli et al., 2023) to improve QFE. Specifically, our main focus is QFE under unawareness of sensitive attributes with multiple classes. To the best of our knowledge, our proposed approach is the first to cover groups with non-binary protected attributes and to estimate ranking fairness across multiple queries. Our main contributions are as follows:

- We propose a new family of principled methods to perform Query Fairness Estimation across multiple queries and non-binary sensitive attributes.
- We introduce the first approach designed to make quantification methods robust against sample selection bias.
- Through extensive experiments on the TREC 2022 Fair Ranking Track collection (Ekstrand, McDonald, Raj, & Johnson, 2022c), we demonstrate that our quantification-based approach outperforms previous methods.

2. Related Work

In recent years, ensuring fairness in search results has emerged as a crucial objective alongside the traditional goal of relevance in the development of IAS (Ekstrand et al., 2022a; Zehlike et al., 2022). To measure the fairness of search results for a given query, i.e., for the task of QFE, several measures have been introduced (Biega et al., 2018; Diaz, Mitra, Ekstrand, Biega, & Carterette, 2020; Kırnap, Diaz, Biega, Ekstrand, Carterette, & Yilmaz, 2021; Kuhlman et al., 2021; Raj & Ekstrand, 2022; Sapiezynski, Zeng, E Robertson, Mislove, & Wilson, 2019; Singh & Joachims, 2018; Yang, Jänich, Mayfield, & Lawrie, 2024; Yang & Stoyanovich, 2017). While these measures cover different notions of fairness, for example *equality of opportunity* (Biega et al., 2018; Hardt, Price, & Srebro, 2016; Diaz et al., 2020; Singh & Joachims, 2018) or *statistical parity* (Geyik et al., 2019; Sapiezynski et al., 2019; Zehlike, Bonchi, Castillo, Hajian, Megahed, & Baeza-Yates, 2017), they all depend on accurate knowledge of the document labels in a ranking. In this work, we consider an "unawareness" scenario, where group labels are unavailable, requiring alternative solutions to ensure accurate fairness assessments.

Related to this, (Ghosh et al., 2021) have conducted an extensive study showing that inferring labels using standard classifiers can be problematic. Their work highlights the need for reliable methods to access accurate fairness labels. Although several studies have focused on enhancing the fairness of classifiers with noisy or incomplete group labels (Celis, Huang, Keswani, & Vishnoi, 2021; Friedler, Scheidegger, & Venkatasubramanian, 2021; Ghosh, Kvitca, & Wilson, 2023; Mozannar, Ohannessian, & Srebro, 2020; Wang, Guo, Narasimhan, Cotter, Gupta, & Jordan, 2020), they primarily address fairness metrics for classification, not for ranking tasks.

In this work, we focus specifically on QFE for rankings under unawareness, an area that has received less attention compared to classification. In the absence of group labels, (Chen & Fang, 2023) proposed a distribution-based learning approach that leverages contextual features. Their approach uses a loss function that does not require explicit group labels but instead targets a fair distribution. Unlike their task of mitigating unfairness, our main objective is to obtain reliable estimates of group proportions in a ranking to accurately measure the fairness of search results.

(Kırnap et al., 2021) proposed a method using a small query-dependent subset of data annotated by human assessors for QFE. Moreover, they only focus on binary group fairness, comparing protected versus non-protected groups. Our work addresses the characteristics of multiclass groups and does not require impractical human annotations for each query.

Related to our work on QFE in rankings, (Ghazimatin, Kleindessner, Russell, Abedjan, & Golebiowski, 2022) have proposed an approach that we term Post-Metric Correction (PMC). Both our approach and the PMC variants include a correction phase and rely on the outputs of an underlying classifier. However, there are significant differences between our quantification-based approaches and the PMC variants. First, the PMC variants are designed only for binary group fairness assessment; our method natively caters to multi-valued sensitive attributes. This is exceedingly rare and important in algorithmic fairness research (Simson, Fabris, & Kern, 2024). Moreover, each PMC variant is tied to a specific independence assumption, which may prove difficult to verify in general settings. Finally, the PMC methods are also tailored to one specific fairness metric, while our approach is more versatile. Our method applies a general pre-correction to the class prevalence estimates, which are then used to compute different fair ranking metrics.

In our work, we propose to use quantification methods to make QFE robust to the limitations when standard classifiers are applied. In a related effort, (Fabris, Esuli, Moreo, & Sebastiani, 2023) have shown that using quantification techniques is a useful way to assess the fairness of algorithms when the labels are unknown. However, their work focuses on classification problems, while our work focuses specifically on QFE.

3. Proposed Approach

3.1 Running Example

Consider an online hiring platform assisting recruiters to fill job openings with promising candidates. Recruiters query the platform with job descriptions and get ranked lists of candidates in return, in decreasing order of estimated job fitness. Platform developers want to ensure that their models are fair with respect to multiple attributes, including age, gender, race, and ethnicity. Given the sensitive nature of this information, most data subjects will be hesitant to disclose it (Bogen, Rieke, & Ahmed, 2020b). Developers, therefore, obtain sensitive attributes for a subset of users through voluntary data disclosure (Wilson, Ghosh, Jiang, Mislove, Baker, Szary, Trindel, & Polli, 2021; LinkedIn, 2024). This incomplete demographic data can be used to estimate platform fairness across all queries and users. Finding the optimal way to perform this estimate is an open research problem tackled in the remainder of this section.

3.2 Learning to Quantify

The field of quantification emerges from the fundamental observation that counting over the labels predicted by a classifier tends to produce poor estimates of class prevalence (see §1.2 in Esuli et al., 2023), unless the classifier is a perfect one. The above naïve counting approach has come to be known as the "Classify & Count" (CC) method (Forman, 2005), and nowadays represents the strawman baseline any proper quantification method is expected to beat.

More formally, a quantifier is a function $\lambda : \mathbb{N}^{\mathcal{X}} \rightarrow \Delta^{n-1}$ mapping bags (or multi-sets) of instances from the input space $\mathcal{X} = \mathbb{R}^d$ to the probability simplex, so that $\lambda(\mathbf{X}) = \mathbf{p}$ lies on the $(n-1)$-simplex defined as $\Delta^{n-1} = \{p_1, \ldots, p_n : p_i \geq 0, \sum_i p_i = 1\}$, in which n is the number of classes $\mathcal{Y} = \{1, \ldots, n\}$ and p_i is the prior probability (a.k.a. "class frequency", or "class prevalence") of class i in bag $\mathbf{X}$. Given a classifier $\phi : \mathcal{X} \rightarrow \mathcal{Y}$, and a bag $\mathbf{X}$, CC is defined as

$$\mathrm{CC}(\mathbf{X})_i = \hat{p}_i = \frac{1}{|\mathbf{X}|} \sum_{\mathbf{x} \in \mathbf{X}} \mathbb{1}[\phi(\mathbf{x}) = i] \tag{1}$$

A caveat on terminology. This paper integrates concepts from different disciplines (quantification, information retrieval, and fairness), each of which employs its own consolidated terminology. Thorough this paper, we will interchangeably use the terms "classes" (here denoted by $\mathcal{Y}$) and "groups" (often denoted by $\mathcal{A}$ in the fairness literature), "labels" and "sensitive attributes". The reader should also note that, despite referring to different concepts, we might interchangeably use "bags" (or "multi-sets") and "ranked lists of items", since our quantifiers regard the latter as unordered objects.

Dataset shift. The essence of quantification is that of tackling a situation in which there is a change ("shift", or "drift") between the distribution P_{tr} from which instances used to train the quantifier have been drawn and the distribution P_{te} from which the test data are drawn. In online hiring, this corresponds to a realistic setting where the training set, consisting of candidates who disclose their sensitive data y, is not sampled IID from the same distribution as the test set. The scenario in which $P_{tr}(X, Y) \neq P_{te}(X, Y)$ is generally known as *dataset shift* (Storkey, 2009).

Although P_{tr} and P_{te} are rather standard notation in machine learning for referring to the training and test distributions, throughout this paper we will consider more than two such distributions. For this reason, we will use the nomenclature P_A to refer to the distribution from which an empirical sample of data items A has been drawn. In this way, we use P_L to denote the distribution from which labelled documents are drawn, and P_U to denote the distribution from which the unlabelled documents are drawn. Further distributions will be introduced when needed.

Among the main types of shift that have been described in the literature, quantification has traditionally focused on *prior probability shift* (PPS). This type of shift is characteristic of *anti-causal learning* (Schölkopf, Janzing, Peters, Sgouritsa, Zhang, & Mooij, 2012) – i.e., learning problems in which the covariates represent symptoms of the phenomenon we want to predict, and that are typically modelled via the factorization $P(X,Y) = P(X|Y)P(Y)$ – and is characterized by the fact that $P_L(Y) \neq P_U(Y)$ while $P_L(X|Y) = P_U(X|Y)$.

A simple quantifier. Arguably, the simplest quantification method devised to counter PPS is the so-called *Adjusted Classify & Count* (ACC) (Forman, 2005). ACC is better described in the binary case $\mathcal{Y} = \{0, 1\}$ (the multiclass extension is straightforward), by observing that

$$\begin{aligned} P_U(\hat{Y} = 1) = {} & P_U(\hat{Y} = 1|Y = 1)P_U(Y = 1) \\ & + P_U(\hat{Y} = 1|Y = 0)P_U(Y = 0) \end{aligned} \tag{2}$$

where $P_U(\hat{Y} = 1)$ corresponds to the CC estimate of the positive class and $P_U(\hat{Y} = 1|Y = 1)$ and $P_U(\hat{Y} = 1|Y = 0)$ are the *true positive rate* (tpr) and *false positive rate* (fpr) of the classifier These two quantities can be estimated using the training data given that the class-conditional distributions of the training and test data are assumed invariant. ACC is thus defined as

$$\mathrm{ACC}(\mathbf{X})_1 = \frac{\mathrm{CC}(\mathbf{X})_1 - \hat{\mathrm{fpr}}}{\hat{\mathrm{tpr}} - \hat{\mathrm{fpr}}} \tag{3}$$

In online hiring, ACC can estimate prevalence of different ethnicities in sets of candidates.

3.3 Countering Sample Selection Bias

Origin. Consider the random variable Q that takes on values 1 ("the item is relevant") and 0 ("the item is irrelevant") with respect to a specific query. Note that the class-conditional distribution is a mixture of relevant and irrelevant items, i.e., $P(X, Q|Y) = P(X|Y, Q = 1)P(Q = 1) + P(X|Y, Q = 0)P(Q = 0)$; however, if U_q are the test documents retrieved for a query, we might expect $P_L(Q = 1) \ll P_{U_q}(Q = 1)$, since it is likely that the vast majority of the training data used for learning our quantifier is irrelevant to a specific query, while the majority of the items retrieved for the query are indeed relevant to it. The class-conditional distributions are thus different, and this clashes with the PPS assumptions.

Note that the random variable Q might be regarded as the "selection variable", which is representative of a type of dataset shift known as *sample selection bias* (SSB).[1] While different quantification methods have shown varying degrees of effectiveness in addressing different types of shift (González, Moreo, & Sebastiani, 2024), we are unaware of any quantification method robust to SSB. In the next section, we propose the first approach to make quantification methods robust against SSB, thereby adapting it for QFE.

Mitigation. Let us turn back to the ACC method (Equation (3)) to illustrate the problem (a similar rationale applies to other quantification algorithms as well, as we discuss in Section 3.4). Recall that ACC replaces $P_U(\hat{Y} = 1|Y)$ with $P_L(\hat{Y} = 1|Y)$ in Equation (2) on the grounds that the class-conditional distributions of training and test datapoints are

1. Sample selection bias is often defined differently, since the selection variable has an effect on the way the training instances (and not the test instances, as in our case) are selected (Storkey, 2009).

invariant. That $P_L(X|Y) = P_U(X|Y)$ implies $P_L(\hat{Y}|Y) = P_U(\hat{Y}|Y)$ follows from the fact that $\hat{Y}$ depends uniquely on X by means of a function (the classifier); inasmuch as the classifier is a measurable function this equivalence holds (Lipton, Wang, & Smola, 2018).

In general, $P_U(\hat{Y} = i|Y = j)$ represents the *classification rates* of the classifier in the *test set*, and is given by

$$P_U(\hat{Y} = i|Y = j) = \mathop{\mathbb{E}}_{\mathbf{x}\sim P_U(X|Y=j)} [\mathbb{1}[\phi(\mathbf{x}) = i]] \tag{4}$$

Of course, we do not have access to the true distribution $P_U(X|Y = j)$ of the expectation, but if we could assume $P_U(X|Y) = P_L(X|Y)$, then this expectation could be estimated by means of an empirical distribution $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim P_L(X|Y = j)$, as

$$P_U(\hat{Y} = i|Y = j) \approx \frac{1}{m}\sum_{k=1}^{m} \mathbb{1}[\phi(\mathbf{x}_k) = i] \tag{5}$$

Although we know that this assumption is flawed in the presence of SSB (Section 3.3), one fundamental observation arises: the pitfall stems from the choice of the empirical distribution used to characterize the classifier ϕ, rather than from the classifier itself.

This is important since most quantifiers use a training set $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, $\mathbf{x}_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$ to both learn a classifier ϕ, *and* estimate, via cross-validation,[2] its classification rates (in our binary example, this reduces to estimating tpr and fpr). Note also that training a classifier is costly, whereas learning the classification rates is rather inexpensive as it only involves issuing predictions and rearranging counts.

Main idea. We propose to disentangle the classifier-training phase from the correction learned by the quantifier. We therefore assume to have access to two sets of labelled data, $L_\phi = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ that we use to train our classifier (offline since it is costly), and $L_{corr} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m'}$ that we use to learn the correction (at query time since it is inexpensive). What remains to make quantification robust to SSB is to counter the selection bias between L_q and L_{corr}.

Figure 1 summarizes our approach, with $\mathbf{M}$ and $\mathbf{t}$ denoting item representations explained Section 3.4. Reducing the sampling bias from the test data is impossible; the test items are retrieved by the search engine precisely to guarantee that items are relevant to a specific query. The main idea we propose in this paper is *to use the same search engine with the same query* to retrieve, from an *auxiliary* pool of training documents L_{corr} (with $L_{corr} \cap L_\phi = \varnothing$), a subset of training items $L_q \subset L_{corr}$ that are biased towards the query similarly to items in U_q. This way, the empirical distribution L_q that we retrieve from the auxiliary pool (L_{corr}) can now be regarded as a sample from a query-biased distribution P_{L_q} and, since we can now assume $P_{L_q}(Q) \approx P_{U_q}(Q)$ (i.e., both distributions are biased towards the query) then we can also assume $P_{L_q}(X, Q|Y) \approx P_{U_q}(X, Q|Y)$ and restore the fundamental PPS assumption. Given that the SSB in L_q mimics the SSB that affects the test data, *the sampling bias shift vanishes.*

We thus consider an auxiliary set of labelled items L_{corr} containing pairs $(\mathbf{x}_i, y_i)$ labelled by sensitive attributes (hereafter called the "correction pool"). From this set L_{corr}, we select, using the same retrieval model and query that we issue on the test pool (U), a ranked list

2. This is in order to avoid the same datapoint being classified to take part in the training of the classifier.

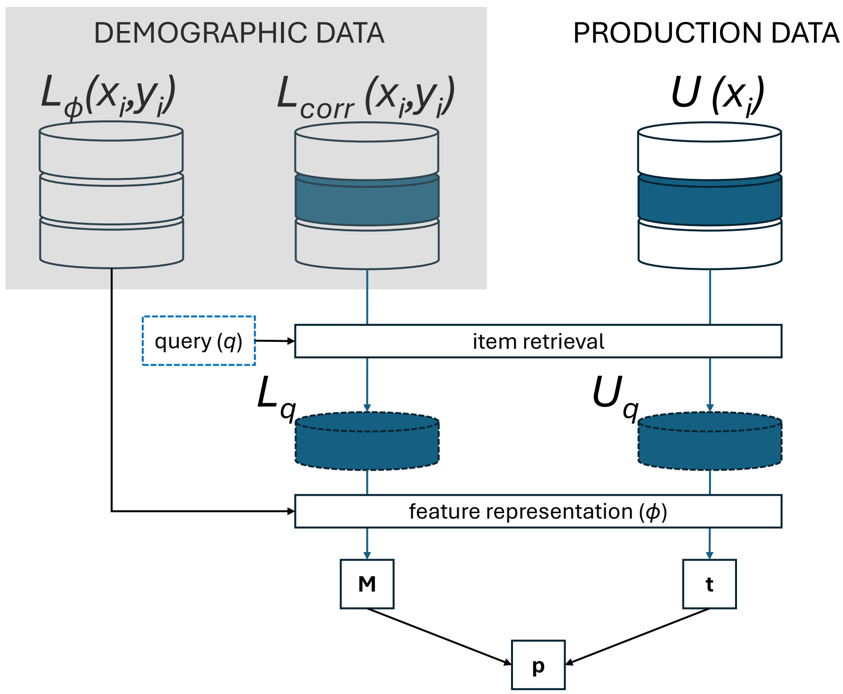

Figure 1: Schematic of our proposed approach for Query Fairness Evaluation. Demographic data is split into L_ϕ, used to learn a feature representation function ϕ, and L_{corr}, employed for query-specific corrections. For each query q, the prevalence $\mathbf{p}$ of sensitive attributes in the retrieved production data U_q is estimated by representing ranked items according to ϕ and leveraging a subset L_q, containing the top-ranked items of L_{corr}, to learn a correction factor $\mathbf{M}$ specific to q.

of (labelled) items L_q that we use to learn a per-query quantification correction to estimate group prevalence in the top-k prefixes of (unlabelled) rankings U_q. Considering ACC applied to online hiring as an example, to estimate the prevalence of different ethnicities in search results U_q for a given query, we compute the classification rates for Equation (3) from a subset $L_q \subset L_{corr}$ of top-ranked items, instead of using the whole labelled set L_{corr}.

3.4 Quantifying Query Fairness

In the previous section, we have explained the intuitions behind our method with respect to (binary) ACC, a relatively simple quantifier. In this section, we generalize the rationale to more sophisticated multiclass quantification methods.

In the modern perspective of multiclass quantification (Bunse, 2022b), most quantifiers can be framed as the problem of solving for $\mathbf{p} \in \Delta^{n-1}$ (unknown prevalences) the system of linear equations

$$\mathbf{t} = \mathbf{M}\mathbf{p} \tag{6}$$

where $\mathbf{t} = \Phi(U)$ is the representation of the test bag U, and $\mathbf{M} = [\Phi(L^1_{corr}), \ldots, \Phi(L^n_{corr})]$ is the matrix containing the class-wise representations of the class-specific correction sets $L^i_{corr} = \{\mathbf{x}_k : (\mathbf{x}_k, y_k) \in L_{corr}, y_k = i\}$, for a given representation function $\Phi : \mathbb{N}^{\mathcal{X}} \rightarrow \mathbb{R}^z$ that embeds bags into z-dimensional vectors, for some z.

Most quantifiers rely on a representation function of the form:

$$\Phi(\mathbf{X}) = \frac{1}{|\mathbf{X}|} \sum_{\mathbf{x} \in \mathbf{X}} \phi(\mathbf{x}) \tag{7}$$

in which a surrogate instance-wise representation function $\phi : \mathcal{X} \rightarrow \mathbb{R}^z$ is invoked, thus effectively computing a mean embedding. Here, we deliberately use ϕ to denote both the representation function and a classifier, since most quantification methods define the representation function upon the output generated by a classifier.

Different choices for Φ and ϕ give rise to different instances of quantification methods. For example, ACC comes down to choosing, as our representation function ϕ, the output of a crisp classifier encoded as a one-hot vector. The columns of $\mathbf{M}$ thus represent the classification rates of the classier (as estimated on correction data), and the problem comes down to reconstructing the class counts of the test examples as a linear combination of , by solving $\mathbf{p} = \mathbf{M}^{-1}\mathbf{t}$.

More sophisticated methods exists. For example, *Probabilistic Adjusted Classify and Count* (PACC) (Bella, Ferri, Hernández-Orallo, & Ramírez-Quintana, 2010) (that we use in our experiments), defines ϕ as a probabilistic classifier returning the posterior probabilities for each class. When $\mathbf{M}$ is not invertible (Bunse, 2022a), the variant we employ frames Equation (6) as the minimization problem

$$\mathbf{p}' = \operatorname*{arg\,min}_{\mathbf{p} \in \Delta^{n-1}} |\mathbf{t} - \mathbf{M}\mathbf{p}|^2 \tag{8}$$

We also use KDEy (Moreo, González, & del Coz, 2024), a state-of-the-art multiclass quantification method that defines Φ as a Gaussian mixture model, obtained via kernel density estimation, of the posterior probabilities returned by a probabilistic classifier. We use the maximum-likelihood variant that solves Equation (6) as the minimization problem

$$\mathbf{p}' = \arg\min_{\mathbf{p}\in\Delta^{n-1}} \mathcal{D}_{\mathrm{KL}}(\mathbf{t}||\mathbf{M}\mathbf{p}) \tag{9}$$

where $\mathcal{D}_{\mathrm{KL}}$ is the well-known Kullback-Leibler divergence.

Note that PACC and KDEy both rely on a probabilistic classifier that is trained in advance on L_ϕ. Equations (8) and (9) only require converting the items in L_q and the test items U_q (both retrieved for the same query but from different pools, L_{corr} and U, respectively) into posterior probabilities and solving the optimization problem. Both operations are rather fast given modern optimization routines (Section 5.4).

4. Experimental Setup

In this work, we aim to provide answers for the following research questions:

- **RQ1**: Does our new method improve over existing baselines?
- **RQ2**: Do quantification techniques allow for accurate QFE in a multiclass setting?
- **RQ3**: To what extent does the performance of our algorithm depend on the size of the correction pool and the rank exposure?

The code that implements all our proposed and all baseline methods, and reproduces our experimental results, is publicly available.[3]

4.1 Dataset & Retrieval

To answer our research questions, we use the TREC 2022 Fair Ranking Track collection (Ekstrand et al., 2022c). This dataset was established for the TREC Fair Ranking Track, in which the fairness of rankings produced by IR systems across different queries and various groups of documents with multiple classes is evaluated. It includes 6.5M English-language Wikipedia articles labelled with group information for various sensitive attributes. We choose this dataset over alternatives from the hiring domain since it is publicly available, it is large, and it encodes several multi-valued sensitive attributes.

To analyse the generalisation of our approach, we select multiple attributes for our evaluation, namely geographic location, gender, and age of topic. Standard classifiers such as logistic regression and SVM demonstrate reasonable accuracy (> 0.75) for these attributes.

In addition to the documents and their demographic labels, the collection includes 97 queries. We index the documents using a Porter stemmer and stop word removal with the help of PyTerrier (Macdonald, Tonellotto, MacAvaney, & Ounis, 2021). As our retrieval model, we use BM25 (Robertson, Walker, Jones, Hancock-Beaulieu, Gatford, et al., 1995) with its standard parameters.

3. `https://github.com/AlexMoreo/query-fairness-estimation`

4.2 Evaluation measures

In our experiments, we assess the accuracy of QFE on multiclass groups. We concentrate on metrics that implement an exposure drop-off based on rank position, thereby reflecting the bias by which users tend to pay more attention to documents ranked higher (Craswell et al., 2008). As our metric of query fairness we rely on the normalized discounted Kullback-Leibler divergence (rKL) (Zehlike et al., 2022) which, for a given set of retrieved documents U_q and different ranking levels $k \in K$ (we consider $K = \{50, 100, 500, 1000\}$), is given by

$$\mathrm{rKL}(U_q) = \frac{1}{Z} \sum_{k \in K} \frac{1}{\log_2 k} \mathcal{D}_{\mathrm{KL}}(\mathbf{p}^k || \mathbf{p}^*) \tag{10}$$

where $\mathbf{p}^k$ is the group distribution for the top-k documents, and $\mathbf{p}^*$ is the group distribution in all judged-relevant documents in the test collection U from which the ranked list U_q is retrieved. Z is simply the normalization factor computed as $Z = \sum_{k \in K} 1/\log_2 k$.

In Section 5.1 we will also test our methods against other baseline methods that are binary-only and that are tailored to one specific fairness metric called normalized discounted difference (rND) (Ghazimatin et al., 2022) defined by

$$\mathrm{rND}(U_q) = \frac{1}{Z} \sum_{k \in K} \frac{1}{\log_2 k} \left| \, p_1^k - p_1^* \, \right| \tag{11}$$

Note that rND only considers the prevalence of the positive class, which is taken to be the prevalence of the protected or disadvantaged group.

Since we work under the unawareness assumption, $\mathbf{p}^k$ is unknown and needs to be estimated. We thus denote by $\mathrm{r\hat{KL}}(U_q)$ (resp. $\mathrm{r\hat{ND}}(U_q)$) the score obtained using predicted distributions $\hat{\mathbf{p}}^k$ in place of the true ones. In order to assess the accuracy on the prediction of the fairness metric M (be it rKL or rND), we report the *absolute error* averaged across all ranked lists U_q retrieved for all queries ($Queries$), which is defined as

$$\mathrm{AE}(Queries, M) = \frac{1}{|Queries|} \sum_{U_q \in Queries} \left| M(U_q) - \hat{M}(U_q) \right| \tag{12}$$

We also report the relative absolute error (RAE), a quantification-specific measure that confronts a predicted distribution with the true distribution, and is defined as

$$\mathrm{RAE}(\mathbf{p}, \hat{\mathbf{p}}) = \frac{1}{n} \sum_{i=1}^{n} \frac{|\hat{p}_i - p_i|}{p_i} \tag{13}$$

We choose RAE as it caters to minority classes by highlighting estimation errors that are small in absolute terms but proportionally large (Sebastiani, 2020).

4.3 Experimental protocol

In this section, we turn to describe the experimental protocol we have designed to provide answers to our RQs. The pseudocode describing our protocol can be consulted in Algorithm 1. The experimental variables we consider are listed below:

Input : • data
• Classifier learner CLS
• Quantification method QUANT
• Group labels

Output: • rKL of the fairness estimates
• RAE of the group prevalence estimates

1 $L, U \leftarrow$ split(data; 50%, 50%)
2 $L_\phi \leftarrow$ draw(L; 500 documents per group)
3 $L \leftarrow L - L_\phi$
4 $\phi \leftarrow \mathrm{CLS}(L_\phi)$
5 **for** $size \in \{10\mathrm{K}, 50\mathrm{K}, 100\mathrm{K}, 500\mathrm{K}, 1\mathrm{M}, 3.25\mathrm{M}\}$ **do**
6     $L_{size} \leftarrow$ undersample(L; $size$)
7     $L_{corr} \coloneqq L_{size}$ # alias
8   **for** query $\in$ *Queries* **do**
9         $U_q \leftarrow$ Retrieve(U, query)@1000
10         $L_q \leftarrow$ Retrieve(L_{corr}, query)@1000
11         $L_q \leftarrow$ keep(L_q; max 200 most relevant documents per group)
12         $\lambda_h \leftarrow$ QUANT(L_q, h)
13   **for** $k \in \{50, 100, 500, 1000\}$ **do**
14             $\hat{\mathbf{p}}^k \leftarrow \lambda_h(U_q^k)$
15             $\mathbf{p}^k \leftarrow$ prevalence(U_q^k)
16   compute RAE($\mathbf{p}^k, \hat{\mathbf{p}}^k$)
17   **end**
18   **end**
19   compute AE(*Queries*, rKL)
20 **end**

Algorithm 1: Experimental protocol.

- *size*: The size of the correction pool L_{corr} from which the documents L_q are retrieved. This is important since there is an evident trade-off between cost and performance: a larger pool size implies higher labelling cost, while at the same time helps in reducing the discrepancy between the distribution of the documents retrieved from the training and test pool (Figure 2). In Line 5 we let *size* vary from a more realistic setting in which 10K labelled documents are assumed available, to the more optimistic (and unrealistic) scenario in which the correction pool is of the same size as the test pool (here corresponding to roughly 3.25M documents). We explore $size \in \{10\mathrm{K}, 50\mathrm{K}, 100\mathrm{K}, 500\mathrm{K}, 1\mathrm{M}, 3.25\mathrm{M}\}$.

- k: The rank cutoff. We investigate the impact the rank of the top-k examined has in the accuracy of QFE. We let k vary in the range $K = \{50, 100, 500, 1000\}$ (Line 13).

- *Queries*: We assess the performance of our methods in QFE across all 97 queries available in the TREC collection (Line 8).

Protocol viewed from the quantification literature

In quantification research, experimental evaluations often use a *sampling generation protocol* to simulate shifts in class distributions, providing a stress test for assessing a quantifier's performance. Typically, these shifts are applied to the test set. In our case, it is not clear how to achieve this without interfering with SSB. Still, we deem it important to impose

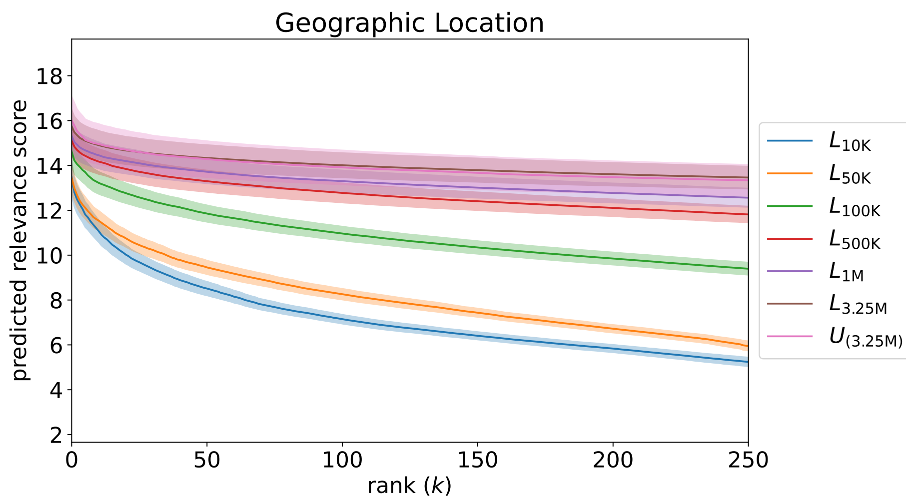

Figure 2: Distribution of predicted relevance score per document rank across all queries. As $|L_{corr}|$ increases its distribution aligns more closely with U.

such a shift in the prior since, in our case, the labelled and unlabelled pools are obtained by partitioning one preexisting collection. This results in the true underlying training and test distributions being nearly identical, which is an oversimplification of the problem. To avoid this oversimplification and test our quantifier under more realistic conditions, we introduce a shift by limiting the labelled data to a fixed number of documents per class (500 for L_ϕ in Line 2, and 200 for L_q in Line 11).

4.4 Methods

We experiment with our proposed variants:

- PACC (Bella et al., 2010): the "Probabilistic Adjusted Classify & Count" quantification method described in Section 3.4. Our proposed variant of PACC models the classification rates matrix of ϕ, using L_q.

- KDEy (Moreo et al., 2024): the KDE-based quantification method described in Section 3.4. Our proposed variant of KDEy models the class-wise densities of the posterior probabilities of ϕ, using L_q.

We compare our methods to the following baselines:

- Naive@k: a method that does not inspect the search results at all to estimate the prevalence of U_q^k but simply reports the prevalence of L_q^k, i.e., of the top-k training documents retrieved for each query from L_{corr}.

- CC (Forman, 2005): the "Classify & Count" method described in Section 3 and Equation (1). This method relies on the predictions issued by ϕ, and does not learn any correction based on L_q.

- PMC_b (Ghazimatin et al., 2022): a binary-only method that corrects a preliminary estimate $\hat{\text{rND}}_\phi$ obtained using “proxy labels” (i.e., labels predicted by ϕ), by applying equation:

$$\text{rND}(U_q) = \frac{\hat{\text{rND}}_\phi}{(1-p)-w} \tag{14}$$

  where $p = P_L(\hat{Y}=1|Y=0)$ and $w = P_L(\hat{Y}=0|Y=1)$ (see Figure 1 (b) in Ghazimatin et al., 2022).

- PMC_b^+: a variant we propose for PMC_b in which the correction factors p and w are not modelled on the same dataset L used to train the classifier (as proposed by the inventors of the method), but from the items L_q retrieved for each query.

- PMC_d (Ghazimatin et al., 2022): a binary-only method that corrects a preliminary estimate $\hat{\text{rND}}_h$ by applying equation:

$$\text{rND}(U_q) = \hat{\text{rND}}_h \cdot \left(\frac{(1-w)\cdot\beta}{x} - \frac{w\cdot\beta}{y}\right) \tag{15}$$

  where p and w are defined as for PMC_b, $\beta = P_L(Y=1)$, and $x = (1-w)\cdot\beta + p\cdot(1-\beta)$ and $y = w\cdot\beta + (1-p)\cdot(1-b)$ (see Figure 1 (d) in Ghazimatin et al., 2022).

- PMC_d^+: a variant we propose for PMC_d in which p, w, β, x, and y are not modelled on L but on L_q.

Our implementations of quantification methods CC, PACC, and KDEy rely on the implementations available in QuaPy (Moreo, Esuli, & Sebastiani, 2021), while the implementations of Naive@k, PMC_b, PMC_b^+, PMC_d, and PMC_d^+ are our own.

Classifier training. All the methods we consider in this work, with the exception of Naive@k, rely on the outputs of a classifier. For the sake of a fair comparison, we use the same classifier in all cases. Our classifier of choice is Logistic Regression (LR), which is arguably becoming a *de facto* choice in the field of quantification, due to the fact that many methods require probabilistic decisions and LR is known to deliver reasonably well-calibrated posterior probabilities (Schumacher, Strohmaier, & Lemmerich, 2023; Moreo et al., 2021). LR is trained offline, once and for all, for each category, on L_ϕ.

Model selection. We perform model selection on the hyperparameters of LR via 5-FCV; we explore the regularization strength $C \in \{10^i\}$ for $-4 \leq i \leq 4$, and the CLASS_WEIGHT parameter in $\{Balanced, None\}$. The only model that has additional hyperparameters is KDEy, which depends on the “bandwidth” of the kernel. Of course, carrying out a model selection phase at query time is not viable. In order to select the bandwidth of KDEy we use 100 queries of the TREC 2021 Fair Ranking Track (Ekstrand, McDonald, Raj, & Johnson, 2022b) collection, a previous version of the TREC 2022 collection. We issue the queries on a 100K-sized correction pool to collect the training documents for all attributes and classes, as well as on U to collect our test rankings U_q. We explored the bandwidth in the range $\{0.01, 0.02, \ldots, 0.10\}$.

Table 1: Results of QFE in terms of AE (lower is better) of rND prediction for $L_{10\text{K}}$ (binary case). KDEy beats all methods.

	Naive@k	CC	PACC (ours)	KDEy (ours)	PMC$_b$	PMC$_b^+$	PMC$_d$	PMC$_d^+$
Geographic Location	$.014^{\dagger} \pm .023$	$.013^{\ddagger} \pm .029$	$.043^{\ddagger} \pm .143$	$\mathbf{.012 \pm .027}$	$.020^{\dagger} \pm .049$	$.030 \pm .087$	$.020^{\dagger} \pm .049$	$.031 \pm .083$
Gender	$.047 \pm .048$	$.014^{\ddagger} \pm .042$	$.016^{\ddagger} \pm .040$	$\mathbf{.011 \pm .026}$	$.014^{\ddagger} \pm .041$	$.015^{\ddagger} \pm .037$	$.014^{\ddagger} \pm .041$	$.015^{\ddagger} \pm .037$
Age of Topic	$.040 \pm .040$	$.025^{\ddagger} \pm .040$	$.019^{\dagger} \pm .023$	$\mathbf{.017 \pm .021}$	$.028 \pm .040$	$.042 \pm .068$	$.028 \pm .040$	$.044 \pm .063$

5. Results

In this section, we report and discuss the experimental results we have obtained. Section 5.1 presents a comparison against the PMC variants (discussed in Section 2); this experimental comparison is discussed separately because the above-mentioned models are binary-only (RQ1). In Section 5.2 we show our main set of experiments, in which we assess the effectiveness of QFE considering multiclass groups (RQ2). Section 5.3 analyses the extent to which the performance of our proposed methods depends on the rank k and the correction pool size (RQ3). Finally, Section 5.4 reports averaged time measurements of our methods.

5.1 Binary protected attributes

In this section, we compare our proposed approach to previous related work (RQ1). The results we discuss in the next paragraphs correspond to the more realistic scenario in which $size = 10\text{K}$. We compare our proposed methods (PACC and KDEy) against the PMC variants (Ghazimatin et al., 2022) Section 4.4.

The PMC methods apply a post-correction to the fairness metric score, making them metric-specific and binary-only. In order to allow for an experimental comparison against the PMC variants, we produce binary versions of our datasets. In the binary setting, the positive class ($Y = 1$) traditionally represents the minority or disadvantaged group. We thus binarize our datasets towards the following groups: "Africa" for Geographic Location, "Female" for Gender, and "Pre-1900s" for Age of Topic. the rest of the groups are merged into the negative class ($Y = 0$).

Table 1 reports the absolute error in the estimation of rND. The displayed values are averaged scores of the absolute error on the prediction of rKL (lower is better) across all 97 queries. Boldface indicates the best method for a given category. Superscripts † and ‡ indicate the methods (if any) whose scores are *not* statistically significantly different from the best one at different confidence levels: symbol † indicates $0.001 < p < 0.01$, while symbol ‡ indicates $p \geq 0.01$. As the test for statistical significance, we rely on the non-parametric Wilcoxon signed-rank test. We use colour coding to facilitate the interpretation of the results, with green indicating the best result and red indicating the worst one per category. From the analysis of the results the following observations can be drawn:

- KDEy is the best-performing approach for the three categories.

- Although CC performs consistently worse than KDEy, the statistical test reveals these differences are not significant. This may be due to high classifier accuracy in the binary case, leaving small room for improvement for the correction phase of other methods. Indeed CC fares significantly worse than KDEy in the multi-class settings (Section 5.2).

- PMC models perform worse than KDEy in a statistically significant sense in most cases. This is an indication that the assumptions upon which PMC models are built do not apply in QFE.

- The variants PMC$^+$ we propose fare consistently worse than the original methods. Intuitively, these variants should perform better, since the documents on which the correction is modelled are more similar to the test data for which the correction is required. This may be an indication that the post correction implemented by the PMC variants do not align with the characteristics of the distributions under consideration in QFE.

- Naive@k performs badly in two out of three cases. This speaks in favour of the ability of KDEy to correct the class prevalence values, since the class prevalence of the top-k training documents is not a good estimate for the top-k test documents *per se*.

- PACC falls short in terms of performance. A plausible reason for this failure is the relatively low number of training documents used to model the classification rates (more on this in Section 5.3).

5.2 Multiclass protected attributes

We now turn to query fairness estimation for multiclass sensitive attributes. Table 2 reports the absolute error of rKL estimates (Equation 10) for a realistic scenario where the dataset size is set to $size = 10K$. Notational conventions are as in Table 1. Since ours is the first multiclass method, we compare our estimator against quantification baselines. The following observations emerge from our results:

- KDEy consistently achieves the best performance compared to all other baselines, with statistically significant differences in the majority of cases across the evaluated attributes. Furthermore, KDEy also exhibits the smallest standard deviation, indicating consistent performance.

- Naive@k performs erratically. In Geographic Location, it achieves results similar to the best-performing method, KDEy. However, in the Gender category, it produces significantly higher errors compared to the best performer. This inconsistent performance is reflected in a high standard deviation across all categories.

- CC performs consistently worse than KDEy in all cases. The differences are statistically significant at $p = 0.01$; in one out of three cases they are significant at $p = 0.001$.

- PACC is, as in the binary case, not competitive with KDEy.

The disparate outcomes we have obtained for PACC and KDEy deserve further analysis. Both methods rely on the same principle of deferring the correction-training phase at query time. As we will see in Section 5.3, though, PACC still performs decently in terms of quantification performance. Concerning RQ2 and in light of our observations, we can conclude that quantification techniques are indeed suitable for accurate QFE in a multiclass scenarios.

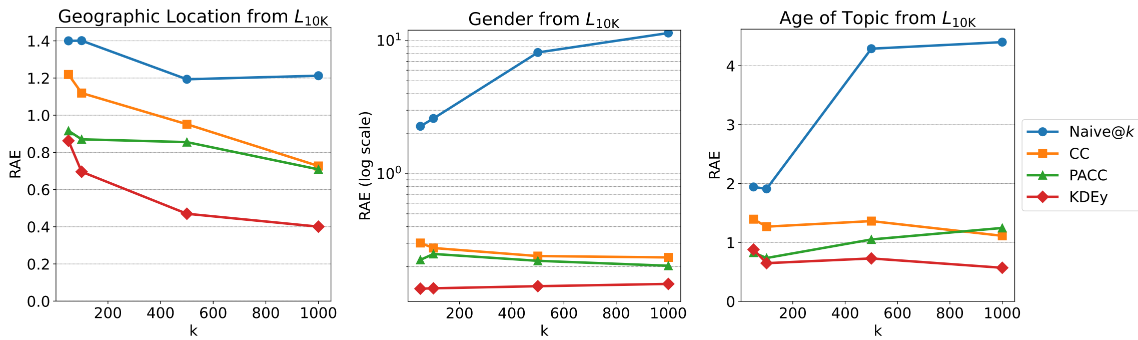

Figure 3: Variations in quantification performance (measured in terms of RAE – lower is better) at different values of k. Note the log-scale in Gender. KDEy and PACC outperform all baselines.

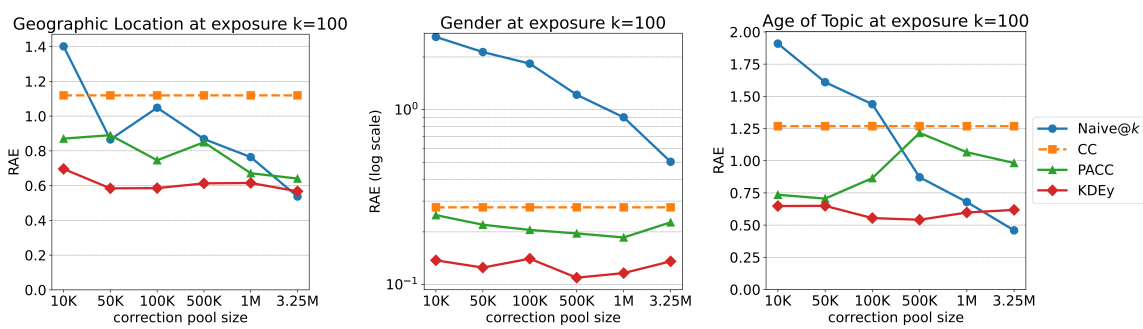

Figure 4: Variations in quantification performance (measured in terms of RAE – lower is better) at different correction pool sizes. Note the log-scale in Gender. KDEy and PACC are more stable than Naive@k and dominate CC.

5.3 Variations of k and size

In this experiment, we analyse the variations in quantification performance at different exposure levels k and variations in the correction pool size (RQ3). We evaluate these experiments in terms of RAE (a quantification-specific measure) between the true distribution and the predicted distribution. Figure 3 displays variations in performance at different rank levels ($k \in \{50, 100, 500, 1000\}$) for the case $L_{corr} \coloneqq L_{10\mathrm{K}}$, while Figure 4 displays variations in performance due to variations in the correction pool size at rank $k = 100$.

Table 2: Results of QFE in terms of AE (lower is better) of rKL prediction for $L_{10\mathrm{K}}$ (multiclass case). KDEy beats all methods.

	Naive@k	CC	PACC (ours)	KDEy (ours)
Geo. Location	.188†± .255	.212 ± .246	.298 ± .241	**.132 ± .181**
Gender	.305 ± .356	.068†± .143	.064 ± .108	**.037 ± .060**
Age of Topic	.213†± .265	.173†± .219	.285 ± .321	**.129 ± .158**

These plots reveal some interesting findings. First, PACC performs consistently better than CC in terms of quantification. This comes as a surprise, since the experiments reported in Table 1 and Table 2 seemed to indicate PACC performs worse than CC.

The key difference with respect to our previous experiments is the evaluation measure under consideration. We argue that RAE is an appropriate measure in multiclass scenarios like the one we are facing here, given its ability to reflect the importance of an error with respect to the proportion of the true prevalence. This can be especially important in cases where one of the classes (the disadvantaged group) tends to display very low prevalence with respect to other classes (especially the privileged group).

However, other evaluation metrics for QFE (e.g., rKL and rND) do not seem to align with the intuitions behind RAE.[4] In the future, it will be interesting to analyse the adequacy of a normalized-discounted-variant of RAE for fairness evaluation. We leave these considerations to future investigations.

Figure 3 shows KDEy performs better than all other methods. Almost all methods show a tendency to improve as the exposure level increases. This is expected, as the quality of an estimated descriptive statistic (in this case: the prevalence) is known to depend on the size of the population under investigation. The only exception to this trend is Naive@k. The reason is that we actively tried to hide the original distribution (not only for Naive@k, but for all methods) by keeping no more than 200 documents per group in L_q (Line 11 in Algorithm 1). Moreover, Naive@k and CC performing consistently worse than PACC and KDEy proves that our quantifiers are effectively learning a correction for the class counts of the classifier which is not spuriously based on the class distribution of L_q.

Figure 4 also shows that Naive@k has a clear dependency on the distributional similarity between L_{corr} and U, as witnessed by its drastic improvement when adding labelled data to the correction pool L_{corr}, which makes it converge towards the same distribution as U. Conversely, the quantification performance of PACC and KDEy is relatively stable, and does not improve markedly when the amount of labelled data available in the pool increases. This indicates that our quantification-based methods are robust to drifts between L_{corr} and U. Moreover, this implies that the amount of labelling effort required to achieve reliable QFE scores is kept under reasonable bounds. Note also that the CC method is represented as a flat curve. The reason is that CC does not leverage the data available in L_{corr} by any means.

5.4 Time Measurements

We measured the training and testing times of our Python implementations on a desktop computer equipped with a 12th Gen Intel(R) i9-12900K processor and 64GB of RAM, running Ubuntu 22. Our methods consist of a per-query correction learning phase followed by a phase of prevalence predictions. On average, PACC required 0.907 ms for learning and 3.316 ms for every prediction, while KDEy took 1.586 ms for learning and 6.395 ms for every prediction. Despite the requirement of a learning phase at query time, our meth-

4. Note that the KL-divergence in rKL also considers the ratio with respect to the target distribution. However, note that it also scales this factor by the predicted prevalence, which might simply be very close to 0, thus cancelling out the term in the aggregation; see Equation (10).

ods demonstrate relatively fast performance and can be integrated into the standard IAS pipeline.

6. Discussion and Conclusion

In this work, we have investigated how to reliably assess the fairness of search results when demographic labels for ranked items are unavailable. We have demonstrated that simply counting over the predictions of a classifier leads to unreliable fairness assessments. To address this limitation, we have proposed novel quantification-based methods to accurately estimate the prevalence of different groups in a ranking. The experimental evaluation has shown that our approach can successfully predict query fairness, including in the previously unaddressed multiclass case, and it does so more accurately than existing methods in the binary case. While most quantification techniques are designed to counter prior probability shift, the problem at hand is instead mainly affected by sample selection bias. To the best of our knowledge, our approach is the first attempt towards making quantification robust to this type of shift naturally occurring in QFE.

Limitations. QFE may be challenging to integrate into learning-to-rank pipelines. First, leveraging protected attributes correction pools comes with infrastructural challenges of data integration. Second, conveying uncertainty of fairness estimates is an important problem we did not address in this work. Third, correction pools may exhibit particular properties (e.g. self-selection effects) that we did not explicitly model.

Future work. In future work, we aim to investigate the suitability of a normalised-discounted variant of RAE for fairness evaluation. We are also interested in exploring different collections where the group prevalence may have naturally varied across training and test conditions. Finally, we aim to endow our QFE solutions with the ability to provide confidence intervals for the estimated values.

Acknowledgments

The work of AE has been supported by the SoBigData.it project (grant IR0000013). The work of FS has been supported by the FAIR project (grant PE00000013). The work of AF has been supported by the Alexander von Humboldt Foundation. The work of AM has been supported by the QuaDaSh project "*Finanziato dall'Unione europea - Next Generation EU, Missione 4 Componente 2 CUP B53D23026250001*". The work of GM and IO has been supported in part by the Engineering and Physical Sciences Research Council grant number EP/Y009800/1, through funding from Responsible Ai UK (KP0011).

References

Adjaye-Gbewonyo, D., Bednarczyk, R. A., Davis, R. L., & Omer, S. B. (2014). Using the bayesian improved surname geocoding method (bisg) to create a working classification of race and ethnicity in a diverse managed care population: A validation study. *Health Services Research*, *49*(1), 268–283.

Bella, A., Ferri, C., Hernández-Orallo, J., & Ramírez-Quintana, M. J. (2010). Quantification via probability estimators. In *Proceedings of the 11th IEEE International Conference* *on Data Mining (ICDM 2010)*, pp. 737–742, Sydney, AU.

Biega, A. J., Gummadi, K. P., & Weikum, G. (2018). Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*, pp. 405–414.

Bogen, M., Rieke, A., & Ahmed, S. (2020a). Awareness in practice: Tensions in access to sensitive attribute data for antidiscrimination. In *Proc, of FAccT*.

Bogen, M., Rieke, A., & Ahmed, S. (2020b). Awareness in practice: Tensions in access to sensitive attribute data for antidiscrimination. In *Proc. of the 3rd ACM Conference on Fairness, Accountability, and Transparency (FAT* 2020)*, pp. 492–500, Barcelona, ES.

Bunse, M. (2022a). On multi-class extensions of adjusted classify and count. In *Proceedings of the 2nd International Workshop on Learning to Quantify (LQ 2022)*, pp. 43–50, Grenoble, IT.

Bunse, M. (2022b). Unification of algorithms for quantification and unfolding. In *Proceedings of the Workshop on Machine Learning for Astroparticle Physics and Astronomy*, pp. 459–468.

Celis, L. E., Huang, L., Keswani, V., & Vishnoi, N. K. (2021). Fair classification with noisy protected attributes: A framework with provable guarantees. In *International Conference on Machine Learning*, pp. 1349–1361. PMLR.

Chen, F., & Fang, H. (2023). Learn to be fair without labels: A distribution-based learning framework for fair ranking. In *Proc. of SIGIR*, pp. 23–32.

Chen, J., Kallus, N., Mao, X., Svacha, G., & Udell, M. (2019). Fairness under unawareness: Assessing disparity when protected class is unobserved. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 339–348.

Chen, L., Ma, R., Hannák, A., & Wilson, C. (2018). Investigating the impact of gender on rank in resume search engines. In *Proceedings of the 2018 chi conference on human factors in computing systems*, pp. 1–14.

Craswell, N., Zoeter, O., Taylor, M., & Ramsey, B. (2008). An experimental comparison of click position-bias models. In *Proc. of WSDM*.

Diaz, F., Mitra, B., Ekstrand, M. D., Biega, A. J., & Carterette, B. (2020). Evaluating stochastic rankings with expected exposure. In *Proc. of CIKM*.

Ekstrand, M. D., Das, A., Burke, R., Diaz, F., et al. (2022a). Fairness in information access systems. *Foundations and Trends® in Information Retrieval*, *16*(1-2), 1–177.

Ekstrand, M. D., McDonald, G., Raj, A., & Johnson, I. (2022b). Overview of the trec 2021 fair ranking track. In *Proc. of TREC*.

Ekstrand, M. D., McDonald, G., Raj, A., & Johnson, I. (2022c). Overview of the TREC 2022 Fair Ranking Track. In *Proc. of TREC*.

Esuli, A., Fabris, A., Moreo, A., & Sebastiani, F. (2023). *Learning to quantify.* Springer Nature, Cham, CH.

European Commission (2024). Regulation on harmonized rules on artificial intelligence (artificial intelligence act). Official Journal of the European Union. Accessed: 2024-08-06.

Fabris, A., Esuli, A., Moreo, A., & Sebastiani, F. (2023). Measuring fairness under unawareness of sensitive attributes: A quantification-based approach. *Journal of Artificial Intelligence Research*, *76*, 1117–1180.

Forman, G. (2005). Counting positives accurately despite inaccurate classification. In *Proceedings of the 16th European Conference on Machine Learning (ECML 2005)*, pp. 564–575, Porto, PT.

Friedler, S. A., Scheidegger, C., & Venkatasubramanian, S. (2021). The (im)possibility of fairness: Different value systems require different mechanisms for fair decision making. *Communications of the ACM*, *64*(4), 136–143.

Geyik, S. C., Ambler, S., & Kenthapadi, K. (2019). Fairness-aware ranking in search and recommendation systems with application to Linkedin talent search. In *Proc. of KDD*.

Ghazimatin, A., Kleindessner, M., Russell, C., Abedjan, Z., & Golebiowski, J. (2022). Measuring fairness of rankings under noisy sensitive information. In *Proc. of FAccT*.

Ghosh, A., Dutt, R., & Wilson, C. (2021). When fair ranking meets uncertain inference. In *Proc. of SIGIR*.

Ghosh, A., Kvitca, P., & Wilson, C. (2023). When fair classification meets noisy protected attributes. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 679–690.

González, P., Moreo, A., & Sebastiani, F. (2024). Binary quantification and dataset shift: an experimental investigation..

Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. *Proc. of NeurIPS*, *29*.

Heuss, M., Sarvi, F., & de Rijke, M. (2022). Fairness of exposure in light of incomplete exposure estimation. In *Proc. of SIGIR*.

Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudik, M., & Wallach, H. (2019). Improving fairness in machine learning systems: What do industry practitioners need?. In *Proc. of CHI*.

Jaenich, T., McDonald, G., & Ounis, I. (2023). ColBERT-FairPRF: Towards fair pseudo-relevance feedback in dense retrieval. In *European Conference on Information Retrieval*, pp. 457–465. Springer.

Jaenich, T., McDonald, G., & Ounis, I. (2024). Fairness-aware exposure allocation via adaptive reranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1504–1513.

Kırnap, Ö., Diaz, F., Biega, A., Ekstrand, M., Carterette, B., & Yilmaz, E. (2021). Estimation of fair ranking metrics with incomplete judgments. In *Proc. of The Web Conference*.

Kopeinik, S., Mara, M., Ratz, L., Krieg, K., Schedl, M., & Rekabsaz, N. (2023). Show me a "male nurse"! how gender bias is reflected in the query formulation of search engine users. In *Proc. of (CHI)*.

Kuhlman, C., Gerych, W., & Rundensteiner, E. (2021). Measuring group advantage: A comparative study of fair ranking metrics. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 674–682.

LinkedIn (2024). Linkedin settings. New York City Council Legislation. Accessed: 2024-08-06.

Lipton, Z. C., Wang, Y., & Smola, A. J. (2018). Detecting and correcting for label shift with black box predictors. In *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*, pp. 3128–3136, Stockholm, SE.

Macdonald, C., Tonellotto, N., MacAvaney, S., & Ounis, I. (2021). PyTerrier: Declarative experimentation in Python from BM25 to dense retrieval. In *Proc. of CIKM*.

Moreo, A., Esuli, A., & Sebastiani, F. (2021). QuaPy: A Python-based framework for quantification. In *Proceedings of the 30th ACM International Conference on Knowledge Management (CIKM 2021)*, pp. 4534–4543, Gold Coast, AU.

Moreo, A., González, P., & del Coz, J. J. (2024). Kernel density estimation for multiclass quantification..

Morik, M., Singh, A., Hong, J., & Joachims, T. (2020). Controlling fairness and bias in dynamic learning-to-rank. In *Proc. of SIGIR*.

Mozannar, H., Ohannessian, M., & Srebro, N. (2020). Fair learning with private demographic data. In *International Conference on Machine Learning*, pp. 7066–7075. PMLR.

New York City Council (2021). Local law 144 of 2021. New York City Council Legislation. Accessed: 2024-08-06.

Pedreschi, D., Ruggieri, S., & Turini, F. (2008). Discrimination-aware data mining. In *Proc. of KDD*.

Raj, A., & Ekstrand, M. D. (2022). Measuring fairness in ranked results: An analytical and empirical comparison. In *Proc. of SIGIR*.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., Gatford, M., et al. (1995). Okapi at TREC-3. *NIST Special Publication Sp*, *109*, 109.

Sapiezynski, P., Zeng, W., E Robertson, R., Mislove, A., & Wilson, C. (2019). Quantifying the impact of user attentionon fair group representation in ranked lists. In *Proc. of Companion Proceedings of WWW*.

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., & Mooij, J. M. (2012). On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML 2012)*, Edinburgh, UK.

Schumacher, T., Strohmaier, M., & Lemmerich, F. (2023). A comparative evaluation of quantification methods..

Sebastiani, F. (2020). Evaluation measures for quantification: An axiomatic approach. *Information Retrieval Journal*, *23*(3), 255–288.

Simson, J., Fabris, A., & Kern, C. (2024). Lazy data practices harm fairness research. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency, FAccT 2024, Rio de Janeiro, Brazil, June 3-6, 2024*, pp. 642–659. ACM.

Singh, A., & Joachims, T. (2018). Fairness of exposure in rankings. In *Proc. of KDD*.

Storkey, A. (2009). When training and test sets are different: Characterizing learning transfer. In Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. D. (Eds.), *Dataset shift in machine learning*, pp. 3–28. The MIT Press, Cambridge, US.

Wang, S., Guo, W., Narasimhan, H., Cotter, A., Gupta, M., & Jordan, M. (2020). Robust optimization for fairness with noisy protected groups. *Advances in Neural Information Processing systems*, *33*, 5190–5203.

Wilson, C., Ghosh, A., Jiang, S., Mislove, A., Baker, L. J., Szary, J., Trindel, K., & Polli, F. (2021). Building and auditing fair algorithms: A case study in candidate screening. In Elish, M. C., Isaac, W., & Zemel, R. S. (Eds.), *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pp. 666–677. ACM.

Yang, E., Jänich, T., Mayfield, J., & Lawrie, D. (2024). Language fairness in multilingual information retrieval. In *Proc. of Companion Proceedings of SIGIR*.

Yang, K., & Stoyanovich, J. (2017). Measuring fairness in ranked outputs. In *Proc. of SSDBM*.

Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., & Baeza-Yates, R. (2017). Fa*ir: A fair top-k ranking algorithm. In *Proc. of CIKM*.

Zehlike, M., Yang, K., & Stoyanovich, J. (2022). Fairness in ranking, part I: Score-based ranking. *ACM Computing Surveys*, *55*(6), 1–36.