# TKB48 at TREC 2021 Fairness Ranking Track

Jin Zhuoqi
Graduate School of Comprehensive Human Sciences, University of Tsukuba
Tsukuba, Ibaraki, Japan
s2021706@s.tsukuba.ac.jp

Hideo Joho
Faculty of Library, Information and Media Science, University of Tsukuba
Tsukuba, Ibaraki, Japan
hideo@slis.tsukuba.ac.jp

Sumio Fujita
Yahoo Japan Corporation
Tokyo, Japan
sufujita@yahoo-corp.jp

## ABSTRACT

Fairness ranking has been recently focused on, which aims to make ranking results fair while keeping relevant. The definition of fairness is diverse. TREC Fairness Ranking Track in 2021 took attention-weighted rank fairness (AWRF) [12] to fit the fairness aspect distribution of ranking results to a population estimator p̂ reflecting the target distribution. TKB48's approach was a post-processing method. We obtained an initial ranking using the BM25 score. We then set a bucket for each of 7 geographic areas in the dataset, and iterated the initial BM25 ranking to choose documents and put them into the bucket in a round-robin manner. As the track evaluated the top 20 results of the final ranking, the goal for us was to make the distribution of each area be the same as the target distribution in the top 20 results. We defined the individual fairness score so that we could choose whether a document should be put into the bucket by comparing an individual fairness score and BM25 score. The individual fairness score was based on how many documents in a certain area has been put into the final ranking. We chose one document with the highest combined score of fairness and relevance for one iterate turn of initial ranking. And we iterated 1000 times so that we could get a final ranking with 1000 documents. Our results ranked fifth out of 13 submissions on the TREC Fairness Ranking Track. Finally, we compared the results of different methods on the TREC Fairness Ranking Track and analyzed it.

## KEYWORDS

Fairness Ranking, Attention-weighted Rank Fairness

## 1 INTRODUCTION

Recently many studies aimed to put fairness into consideration [5,7,8,9]. There are several types of bias that could cause unfairness. The work [5, 6] define them into three types: pre-existing bias, technical bias, and emergent bias. Pre-existing bias includes all biases that exist independently of an algorithm itself [5]. Technical bias arises from technical constraints or considerations [5]. Finally, emergent bias is a bias that searchers pay much attention to items recommended to them or list in the high positions based on their preference, which generates inequity to other items [10].

In this paper, we aim to mitigate unfairness caused by technical bias. The challenge is to reduce unfairness while keep putting items with high relevance in high-rank positions. Pre-processing, in-processing, and post-processing are three main methods to make a fairness ranking [11, 12]. Pre-processing methods mitigate unfairness by handling datasets. In-processing methods mitigate the unfairness during the proceeding of re-ranking while increasing retrieval effectiveness measured by, say, nDCG. In contrast, post-processing methods re-rank the results first and mitigate the unfairness, i.e., re-rank twice. Our method is based on a post-processing method. We used BM25 to get the relevance scores for all documents and then re-rank to fit target distributions for all groups to mitigate unfairness.

The paper is structured as follows: Section 2 discusses advanced work which is related to our work. Section 3 shows our proposed method and dataset in detail. Section 4 analyzes and discusses the results of the experiments. And finally, in section 5 we conclude the paper and with future work.

## 2 RELATED WORK

Singh and Joachims [13] addressed technical bias created by the ranking system. This triggered a direction for fair ranking, which was to reduce the technical bias. In the TREC 2021 fair ranking track, the purpose of task 1 was to design an algorithm to make a fairness ranking, which lay on decreasing technical bias [14]. The method to evaluate fairness is defined in [2], which compares cumulative exposure $\epsilon$ across groups with a population estimator p̂ reflecting the target distribution.

The work [3,4] created pre-processing methods to mitigate biases in the training data, while the work [7, 8] suggested in-processing methods to extend the objective function of a learning-to-rank algorithm by a fairness term. On the other hand, the work [1,9] suggested post-processing methods that assume that a ranking model has already been trained and re-rank the result from the ranking model based on fairness aim.

## 3 METHOD

### 3.1 Decrease distribution difference method (post-processing method)

We focused on geographic fairness. In the dataset, there are seven areas. Some areas have a large number of documents, while some areas only have a few documents. Our aim is to keep the balance of the number of documents from each area in the final ranking. We regard those areas that have few documents as protected groups.

We first used the BM25 score to select 5000 docs as the initial ranking using Solr. After that, we re-ranked the initial ranking based on attention-weighted rank fairness [12]. This compares cumulative exposure across groups with a population estimator p̂ reflecting the target distribution; the system is fairer if the cumulative group exposure is close to the target distribution.

$$AWRF(L) = \triangle(\epsilon(L), \hat{p}) \tag{1}$$

And we used one minus the Jenson-Shannon divergence to compare the exposure.

$$\triangle (P1, P2) = 1 - \frac{1}{2}(D_{KL}(P1|M) + D_{KL}(P2|M)) \tag{2}$$

$$M = \frac{1}{2}(P1 + P2) \tag{3}$$

First, we calculated the target distribution based on the whole dataset. We calculated the number of documents from each area and then normalized them to get the target distribution. Our target is to make the distribution of documents from all areas in the final ranking as similar to the target distribution as possible. Then we started to re-rank the initial ranking with 5000 documents from BM25. Because we want to consider both fairness and relevance, we used the addition of the BM25 score with the individual fairness score as our criterion. We calculated the individual fairness score based on how many documents from the area of the current document still need to be put into the final ranking to fit the target distribution. We used the following formulation to calculate it.

$$individual\_fairness\_score = \frac{T_i}{\sum T_i} \tag{4}$$

Here T means the number of documents that need to be put into the final ranking for each group. And i means the group i, so $T_i$ means the number of documents from the group i need to be put into final ranking. We further calculated a score for each document with the following formulation.

$$score = \alpha * individual\_fairness\_score + (1 - \alpha) * BM25\_score \tag{5}$$

Here $\alpha$ is the parameter to control the weight of two scores. After calculating scores for all documents from the initial ranking, we put one with the highest score to the final ranking and removed it from the initial ranking. Then we renewed T with the following formulation.

$$T_j = T_j - 1 \tag{6}$$

Here $j$ means the one with the highest score we chose from group $j$. However, in the dataset, some documents do not have geographic information. For these documents, we calculated an expectation fairness score with the following formulation.

$$p_i = \frac{C_i}{\sum C_i} \tag{7}$$

$$individual\_fairness\_score = \sum p_i * \frac{T_i}{\sum T_i} \tag{8}$$

$C$ means the number of documents from each group in the current ranking. While current ranking is the documents list after documents which put into the final ranking removed from the initial ranking. And $p_i$ means the probability that documents without geographic information were from group $i$.

### 3.2 Dataset

The corpus we used is from TREC 2021 Fair Ranking Track [14], which consists of articles from English Wikipedia with redirect articles removed and wikitext left intact. The corpus is provided as a JSON file with one record per line.

Each record contains the following four fields:

**id** The unique numeric Wikipedia article identifier.

**title** The article title

**url** The article URL, to comply with Wikipedia licensing attribution requirements

**text** The full article text.

We used id and text fields to do our experiments. We have 57 training topics and 49 test topics from the fair ranking track. The topic is also JSON lines, with each record containing:

**id** A query identifier (int)

**title** The Wikiproject title (string)

**keywords** A collection of search keywords forming the query text (list of str)

**scope** A textual description of the project scope, from its project page (string)

**homepage** The URL for the Wikiproject. This is provided for attribution and is not expected to be used by your system as it will not be present in the evaluation data (string)

**rel_docs** A list of the page IDs of relevant pages (list of int)

Note only training topics have content of rel_docs, which of test topics are empty. Also, we have metadata for most of the docs, which tells us the geographic locations information of each doc so that we can handle geographic fairness to ranking.

## 4 RESULTS AND DISCUSSION

For 57 topics of the training dataset, we extracted geographic information for all relevant documents and calculated the number of documents from each area. We used the distribution of the number of documents from each area as $\hat{p}$ (target distribution). We used Solr to do BM25 retrieval first and got top-5000 results as initial ranking. Then we applied the decrease distribution difference method to re-rank and make the distribution of the number of documents from each area for each query similar to $\hat{p}$. We set $\alpha$ as 0.9. Because of the time limit, this was the only result we submitted to the TREC competition. Figure 1 and Table 1 show the result compared with the best, median, and lowest results of each query.

The result of the submitted method has a higher score than the median result. However, the result of the submitted method is similar to BM25, especially "BM25 with docs which not in geoInfo removed". Because in the submitted method, documents not in geoInfo are removed systematically, and we did not apply normalization to individual fairness score and BM25 relevance score. The ranking largely depended on BM25 relevance scores.

As we did not apply normalization to submitted run, the ranking is greatly dependent on BM25 relevance score so that the result of submitted run is similar to that of BM25. However, the score did not improve after we applied normalization. We speculate the reason should be that we did not get a very good target distribution. A high AWRF@20 of DDD method is based on a great target distribution. So we added human population distribution as a part of the target distribution. AWRF@20 improved a lot which also made the score improve. The results are shown in Table 2 and Table 3.

## 5 CONCLUSION

We designed the decrease distribution difference method to re-rank the result of BM25 to get a fairness ranking, which achieved a better result than the median results of runs from the TREC 2021 Fairness Ranking Track.

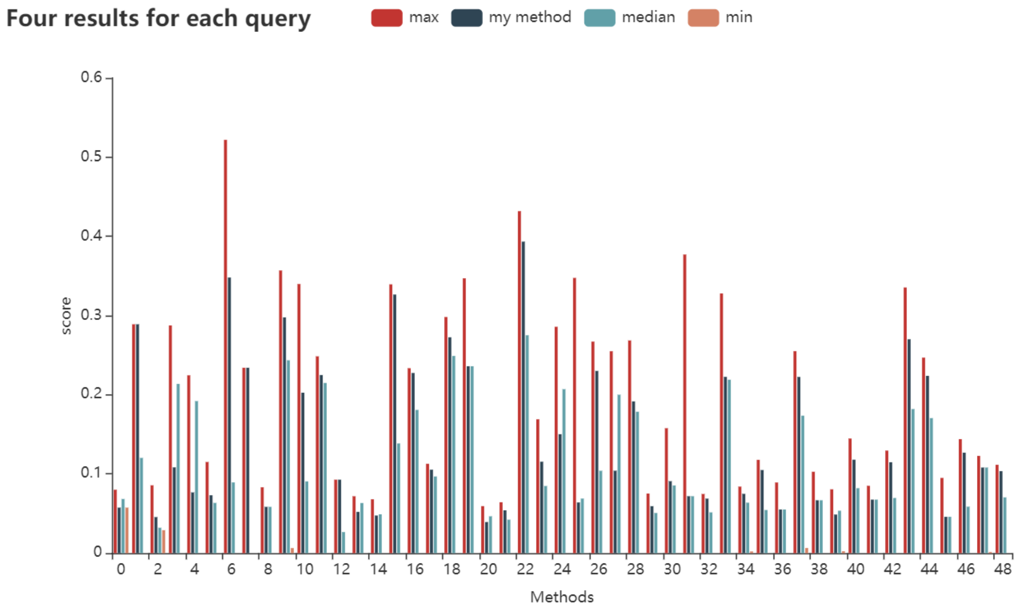

**Figure 1: Four results for each query**

**Table 1: Average results for all queries**

| | max | submitted run | median | min |
|---|---|---|---|---|
| score (AWRF@20*nDCG@20) | 0.199 | 0.143 | 0.111 | 0.002 |

**Table 2: Results for five runs**

| Runs | AWRF@20 | nDCG@20 | score |
|---|---|---|---|
| BM25 | 0.659 | 0.210 | 0.138 |
| BM25 with docs which not in geoInfo removed | 0.663 | 0.216 | 0.142 |
| submitted method (alpha = 0.9) | 0.663 | **0.217** | 0.143 |
| submitted method with normalization (alpha = 0.9) | 0.667 | 0.214 | 0.142 |
| changed target distribution (alpha = 0.9) | **0.676** | 0.214 | **0.144** |

**Table 3: Runs for DDD method with changed target distribution**

| Runs | AWRF@20 | nDCG@20 | score |
|---|---|---|---|
| changed target distribution (alpha = 0.9) | 0.6761 | 0.2143 | 0.1439 |
| changed target distribution (alpha = 0.7) | 0.6762 | 0.2143 | 0.1439 |
| changed target distribution (alpha = 0.5) | 0.6759 | 0.2142 | 0.1438 |
| changed target distribution (alpha = 0.3) | **0.6763** | 0.2143 | **0.1441** |
| changed target distribution (alpha = 0.1) | 0.6726 | **0.2146** | 0.1438 |
| changed target distribution (alpha = 0) | 0.6630 | 0.2165 | 0.1422 |

The limitation of the decrease distribution difference method is the difficulty of getting an accurate target distribution. Different queries could have a different geographic distribution of documents, so the total geographic distribution of documents from the training dataset might not make a good guide for a fair ranking.

Our next step is to apply some learning methods to predict a better target distribution.

## ACKNOWLEDGMENT

Thanks to the organizers of TREC for providing us with the data.

## REFERENCES

[1] Meike Zehlike, Philipp Hacker, and Emil Wiedemann. 2017. Matching code and law: achieving algorithmic fairness with optimal transport. Data Mining and Knowledge Discovery (2017), 1–38.

[2] P. Sapiezynski, W. Zeng, R. E Robertson, A. Mislove, and C. Wilson. Quantifying the impact of user attention on fair group representation in ranked lists. In Companion Proceedings of The 2019 World Wide Web Conference, pages 553–562, 2019.

[3] 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.

[4] Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. 2019. Operationalizing individual fairness with pairwise fair representations. arXiv preprint arXiv:1907.01439 (2019).

[5] Zehlike M, Yang K, Stoyanovich J. Fairness in Ranking: A Survey[J]. arXiv preprint arXiv:2103.14000, 2021.

[6] Batya Friedman and Helen Nissenbaum. 1996. Bias in Computer Systems. ACM Trans. Inf. Syst. 14, 3 (1996), 330–347. https://doi.org/10.1145/230538. 230561

[7] Meike Zehlike and Carlos Castillo. 2018. Reducing disparate exposure in ranking: A learning to rank approach. arXiv preprint arXiv:1805.08716 (2018).

[8] Ashudeep Singh and Thorsten Joachims. 2019. Policy Learning for Fairness in Ranking. arXiv preprint arXiv:1902.04056 (2019).

[9] 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 2017ACM on Conference on Information and Knowledge Management. ACM, 1569–1578.

[10] Bing Pan, Helene Hembrooke, Thorsten Joachims, Lori Lorigo, Geri Gay, and Laura Granka. 2007. In Google we trust: Users' decisions on rank, position, and relevance. Journal of computer-mediated communication 12, 3 (2007), 801–823.

[11] Zehlike M, Castillo C. Reducing disparate exposure in ranking: A learning to rank approach[C]//Proceedings of The Web Conference 2020. 2020: 2849-2855.

[12] P. Sapiezynski, W. Zeng, R. E Robertson, A. Mislove, and C. Wilson. Quantifying the impact of user attention on fair group representation in ranked lists. In Companion Proceedings of The 2019 World Wide Web Conference, pages 553–562, 2019.

[13] Ashudeep Singh and Thorsten Joachims. 2019. Policy Learning for Fairness in Ranking. arXiv preprint arXiv:1902.04056 (2019).

[14] Ekstrand M D, McDonald G, Raj A, et al. TREC 2021 Fair Ranking Track Participant Instructions[J]. 2021.