University of Glasgow Terrier Team at the TREC 2021 Fair Ranking Track

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

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

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

ABSTRACT

In our participation in the TREC 2021 Fair Ranking Track, we investigated novel methods for generating fair rankings that leverage search result diversification and data fusion techniques. In particular, we explore the effectiveness of our approach, that builds on a well-known proportional representation diversification strategy, by experimenting with different inputs to our diversification component to gain insights about how the choices of inputs affect fairness in the generated rankings. To account for different fairness attributes we combine output rankings from our approach with commonly known data fusion techniques, such as CombRank [1], where each output ranking is targeted at a specific fairness attribute. We submitted runs to both of the Fair Ranking tasks (Single Ranking and Multiple Rankings). Our results show that our submitted runs are competitive, especially in the single ranking task, with all runs performing above the TREC-Median in the official track metrics.

1 INTRODUCTION

Our participation in the previous TREC Fair Ranking Tracks have shown that search result diversification can be a promising approach to build on when generating fair rankings [6]. Therefore, in our participation in the 2021 TREC Fair Ranking Track, the University of Glasgow Terrier Team aimed to build on their Terrier.org Information Retrieval platform [4, 7], to further investigate how search result diversification can be leveraged for creating fair rankings strategies. Building on an initial relevance-only ranking, we apply our fairness component that extends a proportional representation based search result diversification approach to generate fair rankings. Moreover, we add a data fusion component to our approach, which has shown to improve diversification in search results [2, 6]. In particular, we participated in both the single ranking and the multiple rankings tasks of the Fair Ranking Track. For the single ranking task, we created a fair ranking approach that exchanges the usual target proportion of a query aspect in proportional representation search results diversification with different inputs, such as the predicted relevance, predicted expected exposure and the fairness attributes of the articles. To try to ensure a fair treatment for multiple fairness attributes, we introduce a data fusion component to combine generated rankings that are individually optimised to maximise the relevance of the ranking and the fairness of exposure for a single fairness attribute. For the multiple rankings task, we build on our single ranking approaches and experiment with different strategies that use knowledge of the previously generated ranking to minimise any disparity of exposure that the fairness attributes receive over the sequence of rankings.

The remainder of this paper is structured as follows: In Section 2, we briefly describe our indexing and relevance-only retrieval strategy. We then present our proposed fairness component in Section 3, before presenting our runs and results in Section 4. We present concluding remarks in Section 5 followed by acknowledgements in Section 6.

2 INDEXING & RETRIEVAL

This section describes the indexing and retrieval of the provided data. The TREC 2021 Fair Ranking Track provided the participants with a corpus of Wikipedia documents, training topics, and metadata corresponding to the documents. We first parsed the document collection to remove the Wiki-Markup. To ensure efficient parsing, we built a python interface to the AutoCorpus Parser called PyAutoCorpus,[1] which resulted in roughly 40x speed-up compared to the existing Wiki-Parsers that we evaluated. After removing the Wiki-Markup, we integrated the TREC 2021 Fair Ranking Track test collection into the well known ir_datasets package [3], which provides access to many useful information retrieval datasets. Having integrated the Fair Ranking test collection into ir_datasets, we could conveniently conduct our experiments within PyTerrier [5]. We have made both PyAutoCorpus and the Fair Ranking test collection integration in ir_datasets publicly available. For our experiments we indexed the collection with stopwords removed and Porter-Stemming applied. We investigated several retrieval strategies and found the PL2 divergence from randomness to be most effective. Therefore, we deploy PL2 for our relevance component in all of our submitted runs, including our relevance-only run.

3 FAIRNESS COMPONENT

To develop our fairness component, we leverage a well known search results diversification approach that is based on proportional representation, i.e., a percentage of the available positions in a ranking are allocated to a particular query aspect based on the distribution of some background population. To leverage this approach for generating fair rankings, we allocate positions in the ranking to different fairness attributes and experiment with different methods for generating the target distributions for allocating rank positions. The generated target distributions are then used as inputs and parameters to our fairness component. In our approach, the protected groups, i.e., the attributes that we aim to be fair to, are derived from the fairness attributes provided by the track organisers. The first fairness attribute is the geographic location of a document, which is explicitly given by the organisers and was available as an attribute in the provided data. A document can contain multiple geographic locations. However, we decided to only use one geographic location per document. The second fairness attribute is only relevant for biographic articles and is described by the track organisers as a demographic attribute. In our participation, we decided to use gender as our demographic attribute since it was

[1] https://github.com/seanmacavaney/pyautocorpus

a reasonable first choice. To infer gender from the article, we used the gender-guesser[2] python library, which predicts gender based on the title of an article. The library is limited in that it can only distinguish between male and female genders. We note that this can be problematic and would hope to use solutions in the future which can account for a more diverse distribution of genders.

We use the following protected groups for the two fairness characteristics. For geographic location we focus on: Africa, Antarctica, Asia, Europe, Latin America and the Caribbean, Northern America and Oceania. For gender, we differentiate between male, female and unknown. To ensure a fair treatment for the groups, we tested different target proportions, all based on the features of the protected groups within our fairness component. For the different geographic groups, we obtained the proportion in the world population[3] and used this as our target proportion. The target proportions for the groups relating to gender were calculated from the proportions within the document collection. Additionally, we also experimented with leveraging the combined predicted relevance scores of the articles from a protected group as a measure of the amount of exposure that a group should receive. For example, we calculated a normalised predicted relevance score for each protected group (geo & gender) and used these scores as our target proportion. Moreover, we also combined the previously calculated proportions for a protected group and the predicted group relevance scores to try to minimise any disparity in exposure that a protected group receives. The following section provides a detailed description of all of our runs and our different methods for generating target proportions.

4 SUBMITTED RUNS AND RESULTS

In this section, we present our submitted runs for Task 1 in Section 4.1 and Task 2 in Section 4.2. We also present an additional unofficial relevance-only run for Task 2 in Section 4.2.

4.1 Task 1: Single Ranking

We submitted five runs for Task 1 consisting of four runs that test different inputs to our diversification method and one relevance-only baseline.

4.1.1 Submitted runs.

- UoGTrDivPropT1: Diversification based on proportions; This run matches the distributions of the protected groups in the generated ranking to their distributions in the background population, i.e., the world population for the geographic attribute and the whole test collection for the gender attribute.
- UoGTrDRelDiT1: Diversification based on relevance distribution; This run allocates positions in the generated ranking to a protected group proportionally with respect to the total relevance scores of the group within the candidate results set.
- UoGTrDExpDisT1: Diversification based on exposure disparity; This run takes into consideration a protected group's distribution in the background collection and the total relevance of the group in the candidate results set.
- UoGTrLambT1: Diversification based on exposure disparity with learned parameter; This run takes into consideration a protected group's distribution in the background collection and the total relevance of the group in the candidate results set. This run differs from our UoGTrDivExpDispT1 run by including a learned parameter. It takes into consideration weights (or importance) of the group relevance and background distributions components.
- UoGTrRelT1: Relevance-only; This run is a relevance-only baseline that has no explicit fairness component.

Runs	NDCG	AWRF	NDCG*AWRF
UoGTrDivPropT1	0.21	0.71	0.15
UoGTrDRelDiT1	0.20	0.80	0.16
UoGTrDExpDisT1	0.20	0.82	0.17
UoGTrLambT1	0.17	0.81	0.15
UoGTrRelT1	0.21	0.65	0.13

Table 1: The table shows the average results over all 49 queries for every submitted run in Task 1. We report the NDCG, the AWRF and the combined metric score. For every metric, the ideal value is 1.

Runs	>Minium	>=Median	= Maximum
UoGTrDivPropT1	48	42	2
UoGTrDRelDiT1	48	48	5
UoGTrDExpDisT1	48	48	24
UoGTrLambT1	48	46	6
UoGTrRelT1	48	40	2

Table 2: The number of queries a run is the minimum, ≥ to the median, or equal to the maximum. Maximum represents the best performing submitted system.

4.1.2 Results. Table 1 shows our results for Task 1. The table reports the average NCDG score as well as the average attention weighted ranked fairness [8] and the product of both, which is the official metric provided by the organizers. The averages are calculated over 49 queries per run.

The table shows, that our run based on exposure disparity, UoGTrDExpDisT1, performs the best of all the runs that we submitted, when averaged over all of the queries. Moreover, we note that the relevance-only run has the worst performance, which is due to it achieving the lowest fairness score from all of our submitted runs. Generally all of our runs achieve a NDCG of approximately 0.20. The highest NDCG score is achieved by our runs that are based on proportion (UoGTrDivPropT1) and relevance-only (UoGTrRelT1). The runs taking relevance into consideration in their fairness component achieve AWRF values around 0.80 (UoGTrDRelDiT1, UoGTrDExpDisT1 and UoGTrLambT1). It is also notable that the submitted run with our learned parameter performs slightly worse than the corresponding run with a parameter set as the default value.

Table 2 reports a per-query analysis of the number of queries for which our runs perform better than the TREC-Minium or TREC-Median, or are equal to the TREC-Maximum performance. We can

[2] https://pypi.org/project/gender-guesser/

[3] https://github.com/fair-trec/trec2021-fair-public/tree/main/data

see from Table 2 that all of our runs are better than the TREC-Minimum in 48 out of 49 queries. For one query, our runs have the lowest overall score in the official metric due to the inability of our retrieval model to deal with the query "1980s" which leads to a NDCG of 0. All of our submitted runs achieve higher scores than the TREC-Median in a majority of the queries. We note that while our relevance-only baseline run is our worst performing run, it manages to perform better than the TREC-Median for 40 out of the 49 queries. Moreover, our run based on exposure disparity (UoGTrDExpDisT1) performs the best from all of the submitted system in 24 out of 49 queries overall submitted runs.

4.2 Task 2: Multiple Rankings

We submitted five runs for Task 2. Four of our runs experiment with different methods for generating distributions for proportional representation, while the fifth run is a relevance-only baseline.

4.2.1 Submitted runs.

- UoGTrDivPropT2: Diversification based on proportions; This run continually optimizes the selection of documents to minimise the divergence, or skew, in the distributions of the protected groups over all of the rankings within a sequence (i.e., for multiple instances of a repeated query), compared to the background population.
- UoGTrDRelDiT2: Diversification based on relevance distribution; This run continually optimises the selection of documents to minimise the divergence, or skew, in the distributions of the protected groups over all of the rankings within the sequence compared to the background population based on the combined predicted relevance scores for all of the articles from a protected group.
- UoGTrDExpDisT2: Diversification based on Exposure Disparity; This run continually optimizes the selection of documents to minimise any disparity between a group's expected and actual exposure.
- UoGTrLambT2: Diversification based on exposure disparity; The approach continually optimises the selection of documents to minimise the disparity between a group's expected and actual exposures. This run differs from our UoGTrDivExpDispT2 run by integrating a parameter to learn the weights, or importance, of the group relevance and background distribution components.
- UoGTrRelT2: Relevance-only; This run was intended to simply consist of a ranking of the documents according to their relevance with respect to the query, for each instance of the query in a sequence. No fairness component is explicitly enforced in this approach. After the submission, we discovered that there was a fault in the code that generated our submitted run for this approach. Therefore, we also report an additional corrected version of the run, denoted as UoGTrRelT2-Fix.

4.2.2 Additional Unofficial Run.

- UoGTrRelT2-Fix: This run is the corrected version of the relevance-only approach for Task 2. This run supersedes the submitted run UoGTrRelT2.

Submitted Runs	EE-L
UoGTrDivPropT2	27.07
UoGTrDRelDiT2	28.48
UoGTrDExpDisT2	28.49
UoGTrLambT2	28.82
UoGTrRelT2	15.65
Additional Unofficial Run:	
UoGTrRelT2-Fix	15.08

Table 3: The table shows the expected exposure loss over all of the queries for each or our submitted runs in Task 2, plus the additional unofficial run. Lower values are better.

Runs	= Minimum	<=Median	=Maximum
UoGTrDivPropT2	0	2	4
UoGTrDRelDiT2	0	2	11
UoGTrDExpDisT2	0	1	10
UoGTrLambT2	0	1	9
UoGTrRelT2	10	8	2
UoGTrRelT2-Fix	10	8	2

Table 4: The number of queries a run achieves the minimum, ≤ the median or the maximum EE-L score. Lower values of EE-L are better, i.e., minimum is the best performing system.

4.2.3 Results. Table 3 shows our results for Task 2. We report the averaged expected exposure loss (EE-L) over all 23 queries (lower values are better). We can see from Table 3 that on average our runs are producing results close to 28.00 EE-L, with the relevance-only baseline being an outlier at 15.00 EE-L.

Table 4 reports the per-query analysis of the number of queries for which our runs achieve the TREC-Minium EE-L, are ≤ the TREC-Median, or are equal to the TREC-Maximum for Task 2. From Table 4, we can see that our runs that include a fairness component, i.e., UoGTrDivPropT2, UoGTrDRelDiT2, UoGTrDExpDisT2 and UoGTrLambT2, achieve the worst EE-L scores from all of the submitted runs for 4, 11, 10 and 9 queries respectively. Moreover, these runs are better than the TREC-Median for only a few queries. Surprisingly, for Task 2, our submitted relevance-only run (UoGTrRelT2) as well as our additional unofficial relevance-only run (UoGTrRelT2-Fix) perform the best out of all of our runs. Notably, both of these runs achieve the TREC-Minimum for 10 out of the 23 evaluated queries.

5 CONCLUSIONS

For our participation in the TREC 2021 Fair Ranking Track we experimented with different approaches for leveraging a search results diversification approach, based on proportional representation, as our proposed fairness component. Moreover, we evaluated different strategies for generating target distributions for allocating positions in a ranking to protected fairness groups. Our results for Task 1 show that search result diversification can be effective for ensuring fairness in ad-hoc rankings. We tested different target proportions as inputs to our approach and found valuable insights about how to optimise for a target exposure. Moreover, we found that our learned parameter in combination with data fusion needs further fine-tuning. For Task 2 we found that our relevance-only baseline performed the best. This provides us with an interesting starting point for further research. Specifically we plan to further investigate how to optimise our fairness components so they do not decrease the relevance of rankings while ensuring fairness of exposure.

6 ACKNOWLEDGMENTS

The authors thank Sean MacAvaney for his valuable input and help during this year's participation.

REFERENCES

[1] Edward A Fox and Joseph A Shaw. 1994. Combination of multiple searches. *NIST special publication SP* 243 (1994).

[2] Shangsong Liang, Zhaochun Ren, and Maarten De Rijke. 2014. Fusion helps diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval.* 303–312.

[3] Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified Data Wrangling with ir_datasets. In *SIGIR*.

[4] Craig Macdonald and Nicola Tonellotto. 2020. Declarative experimentation in information retrieval using pyterrier. In *Proc. of ICTIR*.

[5] Craig Macdonald and Nicola Tonellotto. 2020. Declarative experimentation in information retrieval using pyterrier. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval.* 161–168.

[6] Graham McDonald and Iadh Ounis. 2020. University of Glasgow Terrier Team at the TREC 2020 Fair Ranking Track. (2020).

[7] Iadh Ounis, Gianni Amati, Vassilis Plachouras, Ben He, Craig Macdonald, and Christina Lioma. 2006. Terrier: A High Performance and Scalable Information Retrieval Platform. In *Proc. OSIR at SIGIR*.

[8] Piotr Sapiezynski, Wesley Zeng, Ronald E Robertson, Alan Mislove, and Christo Wilson. 2019. Quantifying the Impact of User Attentionon Fair Group Representation in Ranked Lists. In *Companion Proceedings of The 2019 World Wide Web Conference.* 553–562.