NTCIR 17 Conference: Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies, December 12-15, 2023, Tokyo, Japan
DOI: https://doi.org/10.20736/0002001300

UDInfoLab at the NTCIR-17 FairWeb-1 Task

Fumian Chen
University of Delaware
Newark, DE, USA
fmchen@udel.edu

Hui Fang
University of Delaware
Newark, DE, USA
hfang@udel.edu

ABSTRACT

Providing relevant, diverse, and fair results is crucial for information retrieval systems. It has attracted more and more attention because of issues caused by traditional relevance-centric retrieval systems. These issues include the problem of echo chambers and the increasingly polarized online communities. Therefore, we participated in the NTCIR-17 FairWeb-1 Task to provide group fairness to researchers, movies, and YouTube content and submitted five runs. The runs are based on a recently proposed fair ranking framework, DLF. The experimental results demonstrate that, in many cases, DLF can improve fairness while maintaining relevance but still needs more exploration for ordinal fairness groups and documents with longer text. This paper reports how the runs were constructed and discusses their performance and future work.

KEYWORDS

information retrieval, fair ranking, learning to rank

TEAM NAME

UDInfoLab

SUBTASKS

FairWeb-1

1 INTRODUCTION

The NTCIR-17 FairWeb-1 task [7] was designed to address the problem of unfairness caused by relevance-only retrieval systems. Specifically, besides relevance, FairWeb-1 considers group fairness toward three types of entities, researchers, movies, and YouTube content. Each entity is associated with nominal or ordinal group attributes to define group fairness. The goal is to construct relevant SERPs to given topics and provide fair exposure to each item from different groups. As one of the participants, we were provided with a corpus, Chuweb21D, a pilot data set, and 45 test topics. FairWeb-1 also gives us six baseline runs to save effort for re-ranking-based methods. The five runs we submitted are based on re-ranking these baseline runs with the same re-ranking framework, DLF [2]. In the following sections, we briefly discuss related work and elaborate on our methodology. Then, we explain how the runs were constructed and analyze the results. Finally, we conclude this work and discuss future work.

2 RELATED WORK

Given the importance of fairness in IR systems, countless research attempts have been proposed to provide fair and relevant search results. Among these attempts, learning-to-rank-based methods are more flexible and perform better than traditional methods, such as score-based ones.

Leveraging a fairness-aware loss function combined with relevance utility, DELTR [8], is one of the state-of-art in-processing methods. However, because of the barely available fairness gold labels, it is problematic when using relevance labels as the substitution and training the model using gradient descent. DLF is proposed to solve this limitation and has been tested on the TREC fair ranking dataset [3]. The TREC fair ranking track and FairWeb-1 task have similar corpora, which contain full-text fields of each document. Therefore, we decided to further test DLF's robustness by leveraging it to solve the FairWeb-1 task.

3 METHODS

Given a set of documents $D = (d_1, d_2, ..., d_n)$, or baseline runs to be ranked, we aim to find the best permutation that meets the task's fairness definition and ranks relevant documents at higher positions. Learning-to-rank-based methods first extract features X_i from each document d_i, and then train a scoring function $f(X, \theta) \rightarrow s$ to re-rank the document set.

We re-ranked baseline runs using the recently proposed fair ranking framework, DLF. In this section, we re-state the framework and explain how we deploy the framework on this task.

3.1 Distribution-based learning framework for fair ranking (DLF)

DLF utilizes a fairness-aware loss function and obtains a scoring function for fair ranking by gradient descent, as shown in Equation 1. Given m fairness attributes and n documents to rank:

$$\begin{aligned}\theta^* &= \arg\min FL(\pi) = \sum_{i=1}^{m} w_i * KL(\epsilon_i(\pi), \epsilon_i^*) \\ &= \arg\min \sum_{i=1}^{m} w_i * KL(\sum_{k=1}^{n} P_{\text{fair}}(s_k) * GM_{ik}, \sum_{k=1}^{n} P_{\text{fair}}(s_k^*) * GM_{ik}) \\ &= \arg\min \sum_{i=1}^{m} w_i * KL(\sum_{k=1}^{n} P_{\text{fair}}(f(X_k, \theta)) * GM_{ik}, \epsilon_i^*)\end{aligned} \tag{1}$$

where ϵ^* is the target exposure distribution, KL is the Kullback–Leibler divergence [1], GM is the group membership matrix, and P_{fair} is the top-one fair probability inspired by the original top-one probability [1] and share the same property, such that $P_{\text{fair}}(s_i) = \frac{\phi(s_i)}{\sum_{k=1}^{n} \phi(s_k)}$. DLF assumes a ground truth label s^* exists for fairness yet is barely available. It then uses target exposure distribution to substitute the unavailable ground truth to obtain θ^* as shown in Equation 1. The DLF solves the limitations of previous fair-ranking algorithms in that fairness gold labels are missing, and using relevance labels as substitutions is problematic. The DLF does not solve the limitation of using KL-divergence for ordinal fairness groups, as mentioned

[1] https://en.wikipedia.org/wiki/Kullback-Leibler_divergence

in the FairWeb-1 overview paper. Therefore, this work focuses on nominal fairness groups, such as gender and origin.

According to the DLF framework, we need to merge the scoring function trained at this point with a relevance model to ensure the final ranking is fair and relevant. In this study, we merge the fairness-aware scoring function with baseline models provided by FairWeb-1. The final score, the weighted sum of fairness and relevance scores, will be used to construct our submission.

3.2 Group membership estimation

One of the important components of using DLF is constructing the group membership matrix during training. The matrix reflects the fairness attributes of a document. For example, if a document d_k is from group *male*, then:

$$GM_{\text{gender},k} = \begin{bmatrix} x_{\text{male}} = 1 \\ x_{\text{female}} = 0 \\ x_{\text{non-binary}} = 0 \end{bmatrix}$$

In this task, however, group membership annotation is unavailable, and we decided to extract group membership from the raw HTML for each document. The extraction starts with cleaning the raw HTML to remove special characters and stop words. Then, we utilize *KeyBert* [4] [2] to extract keywords from cleaned HTML text. Last, we leverage the embedding package Sentence-BERT [5] [3] with the pre-trained model 'all-mpnet-base-v2' to embed these keywords and fairness annotation keywords (e.g., "male" and "female" for attribute gender). The group membership is obtained by calculating the cosine similarities between text keyword embeddings and fairness annotation keyword embeddings. For example, if the document keywords have cosine similarity values of 0.8 and 0.2 with male embeddings and female embeddings, respectively, the group membership matrix would be:

$$GM_{\text{gender}} = \begin{bmatrix} x_{\text{male}} = 0.8 \\ x_{\text{female}} = 0.2 \end{bmatrix}$$

Usually, the summation of cosine similarity scores across different fairness groups is not one. In this case, we normalize the scores.

4 EXPERIMENTS

This section discusses how our submitted runs, as shown in Table 1, were constructed in detail. All the runs are based on re-ranking using the DLF framework. To save effort on obtaining initial rankings, re-ranking is based on baseline runs provided by the task organizer. We re-rank five baselines for our submission.

4.1 Dataset and pre-processing

This task adopts the Chuweb21D [4] corpus. The task will be developed and evaluated with three topics: researcher, movie, and YouTube content. For each type of topic, there are 15 test sub-topics.

The only metadata available for each document is the raw HTML. Therefore, we use the Python NLTK [5] package to parse and clean the raw HTML. Then, we follow the processes mentioned in Section 3 to obtain our scoring function. We use the last baseline provided by FairWeb-1 as the training dataset and re-rank the rest for our submission.

[2]https://github.com/MaartenGr/KeyBERT
[3]https://www.sbert.net/
[4]https://github.com/chuzhumin98/Chuweb21D
[5]https://www.nltk.org/

Run	Description
UDinfo-D-RR-1	re-rank run.bm25-D60-D_ver0313.txt leveraging gender and origin-location embeddings
UDinfo-Q-RR-2	re-rank run.bm25-D60-Q_ver0313.txt leveraging gender and origin-location embeddings
UDinfo-D-RR-3	re-rank run.qld-D60-D_ver0313.txt leveraging gender and origin-location embeddings
UDinfo-Q-RR-4	re-rank run.qld-D60-Q_ver0313.txt leveraging gender and origin-location embeddings
UDinfo-D-RR-5	re-rank run.qljm-D60-D_ver0313.txt leveraging gender and origin-location embeddings

Table 1: Submitted runs. We utilize baseline runs provided by the task organizer to save effort in obtaining initial rankings.

4.2 Training feature extraction

Training feature set X plays an important role in model predictive power. Traditionally, features like bm-25 scores and text length are included in the feature set, but they intuitively have less connection with fairness. Therefore, to improve model performance, we need to augment training features. Similar to the process of group membership estimation mentioned in Section 3.2, we use text embedding techniques to obtain our training features. We summarize features extracted in Table 2. As mentioned in Section 3.2, we focus on nominal groups: gender and origin. The feature extraction aligns with our focus, and we leave the ordinal groups for future work.

Features	Description
bm-25	The BM25 score of the topic-document pair
t_gender_sim	The cosine similarity between topic embeddings and gender embeddings
d_gender_sim	The cosine similarity between document embeddings and gender embeddings
t_origin_loc_sim	The cosine similarity between topic embeddings and origin location embeddings
d_origin_loc_sim	The cosine similarity between document embeddings and origin location embeddings
t_gender_sub_sim	The cosine similarity between topic embeddings and gender sub-groups embeddings
d_gender_sub_sim	The cosine similarity between document embeddings and gender sub-groups embeddings
t_origin_sub_sim	The cosine similarity between topic embeddings and origin location sub-groups embeddings
d_origin_sub_sim	The cosine similarity between document embeddings and origin location sub-groups embeddings

Table 2: Summary of Training Features X. Notice that the last four rows listed in the table are four groups of features: the cosine similarity between query/document embeddings and every sub-group (e.g., male, female) embedding, respectively. For example, *q_gender_sub_sim* is actually four features: *q_male_sim, q_female_sim, q_non-binary_sim,* and *q_unknown_sim,* given *gender* has four sub-groups.

4.3 Result and analysis

We first report the relevance performance over 45 topics in Table 4. As can be seen, * indicates improvements over baselines, and for

most of our re-ranking, relevance has been improved over their initial baselines. This shows that DLF preserves relevance when trying to improve fairness.

Run Name	Movie Mean GFR	Researcher Mean GFR	YouTube Mean GFR
UDinfo-D-RR-1	0.4057*	0.4824*	0.3128**
UDinfo-Q-RR-2	0.5956*	0.5064*	0.3428**
UDinfo-D-RR-3	0.3705*	0.5274	0.3285**
UDinfo-Q-RR-4	0.4977*	0.5069	0.3361**
UDinfo-D-RR-5	0.4722*	0.3828	0.3280**
run.bm25-depThre3-D	0.3789	0.4550	0.1733
run.bm25-depThre3-Q	0.4484	0.5064	0.2121
run.qld-depThre3-D	0.3353	0.5389	0.2147
run.qld-depThre3-Q	0.4528	0.5227	0.2453
run.qljm-depThre3-D	0.4456	0.4101	0.2377
run.qljm-depThre3-Q	0.5205	0.4428	0.2024

Table 3: Fairness and relevance improvements of our methods V.S. their baselines over the 15 topics for Movie, Researcher, and YouTube, respectively. * indicates re-ranking improvements over its baseline. ** indicates the improvements are also statistically significant (based on a randomized Tukey HSD test with $B = 5,000$ trials, $\alpha = 0.05$ [6]).

Run	Mean ERR	Run	Mean iRBU
UDinfo-Q-RR-2	0.1788**	UDinfo-Q-RR-2	0.4977*
run.qljm-depThre3-Q	0.1495	UDinfo-Q-RR-4	0.4838*
UDinfo-Q-RR-4	0.1465*	UDinfo-D-RR-3	0.4377**
run.bm25-depThre3-Q	0.1390	run.qljm-depThre3-Q	0.4336
UDinfo-D-RR-5	0.1343*	run.qld-depThre3-Q	0.4330
UDinfo-D-RR-3	0.1310*	UDinfo-D-RR-1	0.4293**
UDinfo-D-RR-1	0.1306*	run.bm25-depThre3-Q	0.4242
run.qld-depThre3-Q	0.1226	UDinfo-D-RR-5	0.4237**
run.qljm-depThre3-D	0.1152	run.qljm-depThre3-D	0.3889
run.qld-depThre3-D	0.1126	run.qld-depThre3-D	0.3872
run.bm25-depThre3-D	0.1113	run.bm25-depThre3-D	0.3624

Table 4: Relevance improvements of our methods V.S. their baselines (mean ERR and iRBU scores for each run over the 15 topics for Movie, Researcher, and YouTube, respectively, in descending order). * indicates that re-ranking improves its baseline. ** indicates the improvements are also statistically significant (based on a randomized Tukey HSD test with $B = 5,000$ trials, $\alpha = 0.05$).

According to Table 5, we display the fairness performance of using the DLF. * also indicates improvements over baselines. For movies, we observed that all re-rankings perform better than their baselines regarding fairness w.r.t. not only origin but also ratings. Given that our algorithm leaves ratings out of the scope, the two fairness categories may have a strong correlation, and improving fairness within one category also helps the other. This might require further exploration, and we leave it for future work. For researcher topics, only two submission runs outperformed their baselines. Even though the algorithm focuses on gender, the runs failed to improve fairness w.r.t. gender. Compared with movies, research papers usually have a longer length and carry more information. Therefore, one possible explanation is that our keyword extraction truncates too much information for research papers and, hence, cannot successfully capture fairness information and improve fairness performance. Even though we did not specifically work on fairness category subscription and YouTube content, most of the submissions this year outperformed the baselines. It might be because diversifying the results list w.r.t. origin and gender helps the result list to be more fair w.r.t. subscription, but this also requires further examination.

Table 3 reports the fairness and relevance combined performance. It shows that in most of the cases for movies and YouTube content, DLF can help improve fairness while maintaining relevance at the same time. However, as mentioned earlier, research papers usually are longer and contain more information. We might need to change how we embed text and calculate similarities to improve model performance for long text accordingly. We leave this part as one of our future work.

5 CONCLUSIONS

This paper discusses our participation in the FairWeb-1 task. We tested a recently proposed fair ranking framework DLF on this movie, researcher, and YouTube content-focused task. The result shows that DLF can help initial rankings improve fairness while maintaining relevance in most cases. For longer text that carries more information, we might need to refine the way for keyword extraction and similarity calculation. Besides, this work leaves ordinal attributes out-of-scope since DLF uses KL-divergence-based loss. The results show that the potential correlations between fairness attributes are also worth exploring. Therefore, in the future, we aim to incorporate DLF with ordinal fairness attributes and explore the relationship between fairness attributes and model performance.

REFERENCES

[1] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*. 129–136.

[2] Fumian Chen and Hui Fang. 2023. Learn to be Fair without Labels: A Distribution-based Learning Framework for Fair Ranking. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*. 23–32.

[3] Michael D Ekstrand, Graham McDonald, Amifa Raj, and Isaac Johnson. 2023. Overview of the TREC 2022 Fair Ranking Track. *arXiv preprint arXiv:2302.05558* (2023).

[4] Maarten Grootendorst. 2020. KeyBERT: Minimal keyword extraction with BERT. https://doi.org/10.5281/zenodo.4461265

[5] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[6] Tetsuya Sakai. 2018. Laboratory experiments in information retrieval. *The information retrieval series* 40 (2018).

[7] Sijie Tao, Tetsuya Sakai, Nuo Chen, Zhumin Chu, Hiromi Arai, Ian Soboroff, Nicola Ferro, and Maria Maistro. 2023. Overview of the NTCIR-17 FairWeb-1 Task. *Proceedings of NTCIR-17. to appear* (2023). https://doi.org/10.20736/0002001318

[8] Meike Zehlike and Carlos Castillo. 2020. Reducing disparate exposure in ranking: A learning to rank approach. In *Proceedings of The Web Conference 2020*. 2849–2855.

Run Name	Movie			Researcher			Youtube	
	Mean GF JSD (Origin)	*Mean GF NMD (Ratings)*	*Mean GF RNOD (Ratings)*	*Mean GF JSD (Gender)*	*Mean GF NMD (H-index)*	*Mean GF RNOD (H-index)*	*Mean GF NMD (Subscription)*	*Mean GF RNOD (Subscription)*
UDinfo-D-RR-1	0.3672*	0.4279*	0.3913*	0.4985	0.4682*	0.4434*	0.3157**	0.3017**
UDinfo-Q-RR-2	0.4493*	0.5132*	0.4706*	0.5096*	0.4977*	0.4605*	0.3315**	0.3081**
UDinfo-D-RR-3	0.3476*	0.3876*	0.3569*	0.5374	0.5195	0.4866	0.3228**	0.3091**
UDinfo-Q-RR-4	0.4601*	0.5161*	0.4750*	0.5190	0.4994	0.4650	0.3309**	0.3157**
UDinfo-D-RR-5	0.4543*	0.4888*	0.4488*	0.3829	0.3765	0.3554	0.3279**	0.3083**
run.bm25-depThre3-D	0.3401	0.3993	0.3630	0.4694	0.4400	0.4155	0.1777	0.1731
run.bm25-depThre3-Q	0.4135	0.4623	0.4283	0.5096	0.4977	0.4605	0.2112	0.2039
run.qld-depThre3-D	0.3122	0.3507	0.3208	0.5497	0.5306	0.4975	0.2155	0.2100
run.qld-depThre3-Q	0.4275	0.4668	0.4351	0.5356	0.5152	0.4807	0.2451	0.2391
run.qljm-depThre3-D	0.4273	0.4606	0.4211	0.4120	0.4038	0.3824	0.2454	0.2329
run.qljm-depThre3-Q	0.4716	0.5462	0.4871	0.4315	0.4362	0.3999	0.2071	0.2038

Table 5: Fairness improvements of our methods V.S. their baselines over the 15 topics for Movie, Researcher, and YouTube, respectively. * indicates that re-ranking improves its baseline. ** indicates the improvements are also statistically significant (based on a randomized Tukey HSD test with $B = 5,000$ trials, $\alpha = 0.05$).