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/10.20736/0002001303

RSLFW at the NTCIR-17 FairWeb-1 Task

Fan Li
Waseda University
Japan, Tokyo
lif@akane.waseda.jp

Kaize Shi
Waseda University
Japan, Tokyo
edmund_kz@akane.waseda.jp

Kenta Inaba
Waseda University
Japan, Tokyo
kinaba@ruri.waseda.jp

Sijie Tao
Waseda University
Japan, Tokyo
tsjmailbox@ruri.waseda.jp

Nuo Chen
Waseda University
Japan, Tokyo
pleviumtan@toki.waseda.jp

Tetsuya Sakai
Waseda University
Japan, Tokyo
tetsuyasakai@acm.org

ABSTRACT

The RSLFW team participated in the NTCIR-17 FairWeb 1 Task. This paper reports our approach to solving the problem and discusses the official results. We applied several different methods to generate 5 runs, including PM-1, PM-2 and DetGreedy algorithm, all of which are post-processing approaches. We also utilized COIL (Contextualize Inverted List) as the RSLFW baseline. By combining official baseline and COIL baseline with different fairness-related algorithm, we analyzed the results of those methods. Our reranked run outperforms the baseline, resulting in an improved GFR score.

KEYWORDS

query expansion, learning to rank, reranking

TEAM NAME

RSLFW

1 INTRODUCTION

The RSLFW team participated in the NTCIR-17 FairWeb 1 Task[4]. This paper reports our approach to solving the problem and discusses the official results.

Our methods can be primarily divided into two directions. The first direction is based on pretrained transformer model called COIL with manual query operation. COIL is one modified BERT-like model that utilizes storing representation vectors into inverted lists. We take the COIL retrieved result as the baseline for RSLFW and employ an election-based approach for fairness-related post-processing. The second direction involves applying Score maximizing greedy mitigation algorithm on the basis of official baseline. We ultimately observe the differences in results between these two directions.

The rest of the paper is divided into four parts as follows. Section 2 introduces related work, and section 3 describes our methods in detail. Results and the reasons behind them are reported in section 4. Finally, section 5 concludes our work.

2 RELATED WORK

Fairness in the field of information retrieval (IR) has been gathering a lot of attentions these days. This concept does not only enables people to reach out relevant documents or pages but also find what would have been dismissed. Learning to rank (LtR) is one of the framework to improve ranked list. Some approaches among LtR have been discussed to introduce fairness into ranked list. One of the approaches is introducing fairness into ranked list after ranked list is built which is called reranking. This approach will make sure to appear protected group in the ranked list. Zehlike et al.[5] suggested reranking method. This research presents a solution for the "Fair Top-k Ranking problem," focusing on selecting a subset of k candidates from a pool of n > k, optimizing utility while adhering to group fairness. The study introduces a fresh definition of ranked group fairness, maintaining protected candidates' proportional representation in the top-k ranking's prefixes. It employs two utility criteria: top-k candidates outperform non-included candidates, and higher-ranked candidates in the top-k surpass their counterparts. The resulting algorithm efficiently generates Fair Top-k Rankings, effectively minimizing biases while maximizing utility, marking a novel advancement in addressing biases in ranked lists.

3 METHODS

This section describes out approach to solving the problem. In the provided dataset, documents lack explicit group labels, complicating the use of group-based approaches. To address this challenge:

For the R topic, we employ Spacy[1] to perform Named Entity Recognition (NER), subsequently assigning labels: female, male, or unknown. This assignment is based on the proportionality of named entities corresponding to each gender. For the M topic, Spacy is once again utilized, this time to extract regional information. The Y topic remains unaltered, with no additional processing.

3.1 Manual Operation of the query

Given that the search results for the M topic should encompass movie-related entities, and those for the Y topic should feature a YouTube video, we made manual adjustments to enhance the accuracy of our search queries. Specifically, for the M topic, we appended the keyword "IMDb" to the end of each query. Similarly, for the Y topic, "YouTube" was concatenated at the query's conclusion. The queries pertaining to the R topic were retained as originally provided.

3.2 COIL Baseline

Recently, information retrieval systems have transitioned from exact lexical matching techniques like BM25 to pretrained transformer models such as BERT for soft semantic matching. However, while the former lacks context sensitivity, the latter compromises computational efficiency. Gao et al.[2] introduced COIL, a novel method

[1] https://spacy.io/

that combines the strengths of both techniques. By encoding document tokens to compute contextualized vector representations with BERT, these vectors are stored in an inverted index corresponding to the token, accompanied by the document id. During a search query, COIL calculates scores using these vector representations, and leverages the special [CLS] token to address vocabulary mismatches. While COIL has outpaced traditional and modern retrieval methods and minimized latency, it struggles with documents exceeding BERT's 512-token limit. To counter this, documents are segmented into 510-token chunks for encoding and then reassembled, ensuring comprehensive representation in the inverted index.

3.3 An Election-based approach

Algorithm 1 PM Algorithm

$s_i \leftarrow 0, \forall i$
for all seats in the ranked list S **do**
  for all aspects $t_i \in T$ **do**
    quotient[i] = $\frac{v_i}{2s_i+1}$
  end for
  $i^* \leftarrow \arg\max_i \text{quotient}[i]$
  if mode = 1 **then**
    $d^* \leftarrow$ pop M_{i^*}
    $s_{i^*} \leftarrow s_{i^*} + 1$
  else if mode = 2 **then**
    $d^* \leftarrow \arg\max_{d_j \in R} \lambda \times \text{quotient}[i^*] \times P(d_j|t_{i^*}) + (1 - \lambda) \sum_{i \neq i^*} \text{quotient}[i] \times P(d_j|t_j)$
    $S \leftarrow S \cup \{d^*\}$
    $R \leftarrow R \setminus \{d^*\}$
    for all aspects $t_i \in T$ **do**
      $s_i \leftarrow s_i + \frac{P(d^*|t_i)}{\sum_{t_j} P(d^*|t_j)}$
    end for
  end if
end for

Dang and Croft[1] introduce an innovative concept in the realm of search results diversification: "diversity by proportionality." Contrary to traditional diversification techniques, they assert that a result list achieves optimum diversity when it furnishes documents related to a query's associated topics in numbers proportionate to each topic's prevalence. The inspiration for their framework stems from electoral systems where seats are distributed among political parties based on the votes they garner.

Drawing from the Sainte-Laguë Method used in electoral contexts, they introduced PM-1. In this method, each spot on the search result list is assessed iteratively. For each position, PM-1 computes a quotient for all topics based on their relative popularity, and the topic with the highest quotient is selected. The top-ranking document for that topic then claims that position in the list. In PM-1, represented by *choose mode=1*, v_i and s_i denote the expected number of documents for aspect t_i and the count of documents already assigned to t_i respectively. The set M_i is comprised of documents $\{d(1), d(2), \ldots, d(l_i)\}$ related to aspect t_i, with the top document for t_i being the document with higher relevance score in M_i.

Building on this concept, a probabilistic interpretation of the Sainte-Laguë Method, known as PM-2, further refines this approach by accommodating the likelihood that a document might pertain to multiple topics simultaneously. PM-2 operates on the assumption that all documents d_j in D are relevant to every aspect t_i in T, each having a relevance probability $P(d_j|t_i)$. When identifying the next optimal document, the factor λ balances relevance to the prime aspect t_i^* against broader aspect relevance. Unlike PM-1, where s_i denotes the count of seats occupied by t_i, in PM-2, s_i is better perceived as the proportion of the seat that t_i occupies, since a selected document d^* is presumed relevant to all aspects and hence multiple aspects share the seat.

We believe that both these approaches are suitable for our fair ranking problem, where we expect different groups can secure their supposed-to-be "seats" in a document ranking. The details of the algorithm are elaborated in Algorithm 1.

3.4 DetGreedy-based approach

DetGreedy is proposed by LinkedIn[3] for mitigating bias in search and recommendation systems. The core idea of DetGreedy algorithm is to prioritize satisfying minority groups. For the different attributes in sorting, a range of feasible sorting algorithms is defined as shown below:

$$\forall k \leq |\tau_r| \,\&\, \forall a_i \in A,\ \text{count}_k(a_i) \leq \lceil p(a_i) \cdot k \rceil \text{ and,} \quad (1)$$

$$\forall k \leq |\tau_r| \,\&\, \forall a_i \in A,\ \text{count}_k(a_i) \geq \lfloor p(a_i) \cdot k \rfloor \quad (2)$$

where k is number of desired results, τ_r represents ranked list of candidates for search request r, and A is set of fairness attributes to consider. $p(a_i)$ is desired distribution of attribute a_i.

A feasible fairness ranking algorithm ensures that the protected attributes have lower and upper limits, referred to as minimum representation and maximum representation. Once ranking list starts to appear that the current number of attributes is less than the minimum representation or greater than the maximum representation, it is considered to be infeasible.

DetGreedy algorithm performs cumulative counting analysis on each row of the initial ranking result. When the value of current attribute violates the minimum representation requirement, the algorithm selects the attribute with the highest relevance score from those attribute clusters which are less than minimum representation. If current attribute value meets minimum representation, the algorithm then selects the attribute value with the highest relevance score from the clusters that have not met the maximum representation. Algorithm 2 shows the detailed information of DetGreedy.

In contrast, DetConstSort algorithm, which is also introduced by LinkedIn[3], incrementally increases the representation of attribute values and then inserts candidates based on their scores. Specifically, DetConstSort traverses through the results from the beginning until a attribute value meeting the minimum representation is found. The next candidate with that attribute value is selected. If inserting this candidate violates the maximum representation, the algorithm attempts to swap this candidate to a position that satisfies the maximum representation requirement, while maintaining a higher score for results in earlier positions.

Through extensive experimentation, it has been demonstrated that the DetGreedy algorithm performs the best in terms of fairness,

Algorithm 2 Score Maximizing Greedy Mitigation Algorithm (DetGreedy)

foreach $a_i \in a$, counts$[a_i] := 0$
rankedAttList := []; rankedScoreList := []
for $k \in \{1, \ldots, k_{\max}\}$ **do**
  belowMin := $\{a_i : \text{counts}[a_i] < \lfloor k \cdot p_a \rfloor\}$
  belowMax := $\{a_i : \text{counts}[a_i] \geq \lfloor k \cdot p_{a_i} \rfloor$ and counts$[a_i] < \lceil k \cdot p_{a_i} \rceil\}$
  if belowMin $\neq \emptyset$ **then**
    nextAtt := $\arg\max_{a_i \in \text{belowMin}} s_{a_i,\text{counts}[a_i]}$
  else
    nextAtt := $\arg\max_{a_i \in \text{belowMax}} s_{a_i,\text{counts}[a_i]}$
  rankedAttList[k] := nextAtt
  rankedScoreList[k] := $s_{\text{nextAtt,counts[nextAtt]}}$
  counts[nextAtt]++
return [rankedAttList,rankedScoreList]

provided strict minimum representation requirements are not imposed for every attribute. Consequently, we choose the DetGreedy algorithm as our baseline-based reranking method.

4 EXPERIMENTS

4.1 Run Description

The RSLFW team contributed five runs for each topic category. Of these, two were designated as 'RR' runs, signifying the application of a reranking algorithm on the official run. The other three were reranked based on the COIL baseline. For the COIL baseline construction, we took inspiration from Gao et al.'s work and utilized BERT-base (uncased version with 768 CLS dimensions and 110M parameters) as our pretrained language model. This model was trained using the MSMARCO passage dataset, a compilation of user queries sourced from Bing's search logs combined with passages from web documents. To ensure computational efficiency with COIL, we first curated a sub-corpus, comprising the top 1,000 most relevant documents for each topic, using the BM25 algorithm implemented in Anserini. The details of each run are outlined in Table 1.

Table 1: Description of RSLFW Team's Runs

Run ID	Baseline	Reranking Algorithm
RSLFW-Q-MN-1	COIL	None
RSLFW-Q-MN-2	COIL	PM1
RSLFW-Q-MN-3	COIL	PM2
RSLFW-Q-RR-4	Official	PM2
RSLFW-Q-RR-5	Official	DetGreedy Algorithm

4.2 Results

4.2.1 Evaluation results. Based on the results for the M topic presented in Table3, a comparison between RSLFW-Q-MN-1 and its counterparts, RSLFW-Q-MN-3 and RSLFW-Q-MN-2, reveals that RSLFW-Q-MN-1 consistently scores higher in both ERR and iRBU metrics. This suggests that the run without reranking is more relevant. Conversely, RSLFW-Q-MN-3 and RSLFW-Q-MN-2 consistently achieve superior group fairness scores. This pattern aligns with the commonly understood "relevance and fairness trade-off."

In the context of reranked runs, RSLFW-Q-RR-4 and RSLFW-Q-RR-5, both based on the official run, exhibit distinct performances. Despite the fact that both of them show no statistically significant different from each baseline, RSLFW-Q-RR-5 demonstrates a superior relevance score. On the other hand, RSLFW-Q-RR-4 excels in the group fairness score, earning the highest GFR score among all submitted runs as shown in Table3. Comparing RSLFW-Q-RR-5 and its origin, baseline, our reranking method slightly improves most of all the scores. The reranking method also improves the relevance scores as shown in Table2.

For the R topic, as depicted in Tables4, the results bear a resemblance to those of the M topic. The exception is RSLFW-Q-RR-5, which emerges as the top performer in the GFR score among all the submitted runs.

However, the Y topic results, presented in Rable5, deviate slightly. Here, RSLFW-Q-MN-3 and RSLFW-Q-MN-2 outperform RSLFW-Q-MN-1 in both relevance and group fairness scores. Moreover, RSLFW-Q-RR-4 secures the leading position for the GFR score among all the submitted runs.

Table 2: Evaluation results over the 45 topics for our submitted runs. The baseline is run.bm25-depThre3-Q

Run	ERR	iRBU
RSLFW-Q-MN-1	0.1245	0.3510
RSLFW-Q-MN-2	0.1067	0.3298
RSLFW-Q-MN-3	0.0770	0.2271
RSLFW-Q-RR-4	0.1485	0.4737
RSLFW-Q-RR-5	0.1847	0.4973
baseline	0.1390	0.4242

Table 3: Evaluation results over the 15 M topics for our submitted runs. The baseline is run.bm25-depThre3-Q

Run	ERR	iRBU	GFR
RSLFW-Q-MN-1	0.1893	0.4268	0.3581
RSLFW-Q-MN-2	0.1489	0.4250	0.3756
RSLFW-Q-MN-3	0.1489	0.4250	0.3756
RSLFW-Q-RR-4	0.1620	0.5463	0.4996
RSLFW-Q-RR-5	0.2044	0.5674	0.4949
baseline	0.1712	0.5035	0.4484

4.2.2 Topic Analysis. We conducted a per-topic analysis for further discussion. The analysis is based on GFR, and the Bronze-All file was used as the relevance assessment. To prevent redundancy, from this point forward, RSLFW-Q-RR-5 is abbreviated as Run 5 and run.bm25-depThre3-query as baseline, and so on. Table 6 shows the topics and GFR scores where RSLFW-Q-RR-5 significantly outperformed the baseline. The topic with the largest difference between two runs was "car racing movies." Only five topics' scores decreased and the mean difference is -0.014. The reranking method does not appear to have a negative impact.

Table 4: Evaluation results over the 15 R topics for our submitted runs. The baseline is run.bm25-depThre3-Q

Run	ERR	iRBU	GFR
RSLFW-Q-MN-1	0.1021	0.3700	0.3622
RSLFW-Q-MN-2	0.0890	0.3084	0.3027
RSLFW-Q-MN-3	0.0000	0.0000	0.0000
RSLFW-Q-RR-4	0.1478	0.4974	0.4807
RSLFW-Q-RR-5	0.2131	0.5488	0.5010
baseline	0.1989	0.5489	0.5064

Table 5: Evaluation results over the 15 Y topics for our submitted runs. The baseline is run.bm25-depThre3-Q

Run	ERR	iRBU	GFR
RSLFW-Q-MN-1	0.0822	0.2562	0.2381
RSLFW-Q-MN-2	0.0822	0.2562	0.2381
RSLFW-Q-MN-3	0.0822	0.2562	0.2381
RSLFW-Q-RR-4	0.1357	0.3775	0.3428
RSLFW-Q-RR-5	0.1365	0.3757	0.3408
baseline	0.0471	0.2202	0.2121

Table 6: GFR of the topics where Run 5 significantly outperformed the baseline

topic	query	Run 5	baseline	difference
M007	car racing movies	0.4581	0.0000	0.4581
Y001	Bacharach/David covers	0.4439	0.0000	0.4439
Y002	Beatles covers	0.4001	0.0432	0.3569

5 CONCLUSIONS

The RSLFW team participated in the NTCIR-17 Fair Web(FairWeb-1) task. We submitted five runs. Two of them are based on COIL and the others are reranked on the official baseline. We discussed the trade-offs between relevance and fairness. When comparing our reranked run with the baseline, we observed an improvement in the GFR score.

REFERENCES

[1] Van Dang and W. Bruce Croft. 2012. Diversity by proportionality: an election-based approach to search result diversification. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. https://api.semanticscholar.org/CorpusID:11318288

[2] Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. COIL: Revisit Exact Lexical Match in Information Retrieval with Contextualized Inverted List. arXiv:2104.07186 [cs.IR]

[3] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. 2019. Fairness-Aware Ranking in Search & Recommendation Systems with Application to LinkedIn Talent Search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2221–2231. https://doi.org/10.1145/3292500.3330691

[4] Sijie Tao, Nuo Chen, Tetsuya Sakai, Zhumin Chu, Nicola Ferro, Maria Maistro, Ian Soboroff, and Hiromi Arai. 2023. Overview of the NTCIR-17 FairWeb-1 Task. In *NTCIR-16 to appear*.

[5] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. FA*IR: A Fair Top-k Ranking Algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (Singapore, Singapore) *(CIKM '17)*. Association for Computing Machinery, New York, NY, USA, 1569–1578. https://doi.org/10.1145/3132847.3132938