Was Fairness in IR Discussed by Cooper and Robertson in the 1970's?

Djoerd Hiemstra
Radboud University
The Netherlands
hiemstra@cs.ru.nl

Abstract

I discuss fairness in Information Retrieval (IR) through the eyes of Cooper and Robertson's probability ranking principle. I argue that unfair rankings may arise from blindly applying the principle without checking whether its preconditions are met. Following this argument, unfair rankings originate from the application of learning-to-rank approaches in cases where they should not be applied according to the probability ranking principle. I use two examples to show that fairer rankings may also be more relevant than rankings that are based on the probability ranking principle.

1 Introduction

Like many people, I love to do an "ego search" in Google[1], to see what comes up when I search my name. When Latanya Sweeney did an ego search about a decade ago, she was shocked to find advertisements for background checks with the headline "Latanya Sweeney, Arrested?" Sweeney, professor at Harvard, never was arrested. One of her colleagues suggested that the advertisement came up because of her "black name" – Latanya is a popular name among Americans of African descent. In other words, the advertisement ranking algorithm was racist. Motivated by this incident, Sweeney [2013] investigated the Google results for more than 2,000 racially associated personal names, and showed that Google's advertisements are indeed systematically racially biased. Sweeney's work was pivotal in putting bias and fairness of algorithms on the global research agenda.

The harm that (search) algorithms may do is substantial, especially if the algorithms are opaque, and if clicks on the (racist, unfair) results are fed back into the algorithm, thereby creating a destructive feedback loop where clicks on unfair results further reinforce the system's unfairness. Cathy O'Neil compared such algorithms to weapons of mass destruction, because their destruction scales to hundreds of millions of (Google) users. O'Neil [2016] wittingly called her book, which I highly recommend, Weapons of Math Destruction.

In this paper, I look at two motivating examples of fair/unfair rankings, formulated 45 years apart, even though the term *fairness* was not used in the early example.

[1] I use Ecosia for my other searches.

2 Fairness in Dynamic Learning to Rank

Let's first discuss fairness by following the example of Morik et al. [2020], who were awarded the best paper at SIGIR 2020. They present the following motivating example:

> Consider the following omniscient variant of the naive algorithm that ranks the articles by their true average relevance (i.e. the true fraction of users who want to read each article). How can this ranking be unfair? Let us assume that we have two groups of articles, G_{right} and G_{left}, with 10 items each (i.e. articles from politically right- and left-leaning sources). 51% of the users (right-leaning) want to read the articles in group G_{right}, but not the articles in group G_{left}. In reverse, the remaining 49% of the users (left-leaning) like only the articles in G_{left}. Ranking articles solely by their true average relevance puts items from G_{right} into positions 1–10 and the items from G_{left} in positions 11–20. This means the platform gives the articles in G_{left} vastly less exposure than those in G_{right}. We argue that this can be considered unfair since the two groups receive disproportionately different outcomes despite having similar merit (i.e. relevance). Here, a 2% difference in average relevance leads to a much larger difference in exposure between the groups.
> (Morik et al. [2020])

This example clearly shows a problem with fairness since right-leaning users have all their preferred documents ranked before the documents that are preferred by left-leaning users. Documents from the minority group (left-leaning in the example) are never even shown on the first results page. The example furthermore suggests that the ranking is optimal given the "true relevance" of the items, but is it really? Let's have a look at some well-known evaluation measures for the ranking presented in the example, and for a fairer ranking where we interleave right-leaning and left-leaning documents, starting with a right-leaning document.

Ranking Algorithm	RR	AP	nDCG
relevance ranking (unfair)	0.55	0.59	0.78
interleaved ranking (fair)	0.76	0.45	0.78

Table 1: Evaluation results for Morik's example (Morik et al. [2020])

Table 1 shows the expected evaluation results if, as stated in the example, 51% of the users like the right-leaning documents and 49% of the users like the left-leaning documents. For instance, the expected reciprocal rank (RR) for the relevance ranking in the example is 0.51 times 1 (51% of the users are satisfied with the first result returned) plus 0.49 times 1/11 (49% of the users are dissatisfied with the first ten results, but satisfied with the eleventh result). The table also shows expected average precision (AP) and the normalized discounted cumulative gain (nDCG). So, if we are interested in the rank of the first relevant result (RR), then the example ranking is not only unfair, it is also of lower overall quality. If we are more interested in recall as measured by AP, then the relevance ranking indeed outperforms the interleaved ranking. Finally, in case of nDCG, the results are practically equal (the relevance ranking outperforms the interleaved ranking in the third digit). NDCG is normally used in cases where we have grades of relevance judgments, e.g. grade 2 for *very relevant* and grade 1 for *marginally relevant*. If we additionally assume that the top ranked right-leaning document and the top ranked left-leaning document are more relevant (relevance score 2) than the other relevant documents (relevance score 1), then the fair, interleaved ranking outperforms the unfair, relevance ranking: 0.78 vs. 0.76. So, depending on our evaluation measures, the ranking by the "true average relevance" might not produce the best quality search engine. It clearly produces an unfair search engine.

3 Fairness in Probabilistic Retrieval

Interestingly, rankings where two groups of users prefer different sets of documents were already discussed more than 45 years ago by Stephen Robertson when he introduced the probability ranking principle. Robertson [1977] contributed the principle to William Cooper. The paper's appendix contains the following counter-example to the probability ranking principle, which Robertson also contributed to Cooper. The example follows the above example closely, but with different statistics for the two groups of users:

> Cooper considers the problem of ranking the output of a system in response to a given request. Thus he is concerned with the class of users who put the same request to the system, and with a ranking of the documents in response to this one request which will optimize performance for this class of users. Consider, then, the following situation. The class of users (associated with this one request) consists of two subclasses, U_1 and U_2; U_1 has twice as many members as U_2: Any user from U_1 would be satisfied with any one of the documents D_1–D_9, but with no others. Any user U_2 would be satisfied with document D_{10}, but with no others. Hence: any document from D_1–D_9, considered on its own, has a probability of 2/3 of satisfying the next user who puts this request to the system. D_{10} has a probability of 1/3 of satisfying him/her; all other documents have probability zero. The probability ranking principle therefore says that D_1–D_9 should be given joint rank 1, D_{10} rank 2, and all others rank 3. But this means that while U_1 users are satisfied with the first document they receive, U_2 users have to reject nine documents before they reach the one they want. One could readily improve on the probability ranking, by giving D_1 (say) rank 1, D_{10} rank 2, and D_2–D_9 and all others rank 3. Then U_1 users are still satisfied with the first document, but U_2 users are now satisfied with the second. Thus the ranking specified by the probability-ranking principle is not optimal. Such is Cooper's counter-example.
> (Robertson [1977])

Let's again look at the evaluation results for the rankings presented in the example, the relevance ranking and the improved ranking, which we indicate as above as interleaved.

Table 2 shows that the relevance ranking, that ranks all documents preferred by users from group U_1 above those preferred by users from group U_2 treats the minority group U_2 unfairly. It also produces lower quality results than the interleaved ranking on all three evaluation measures. But why would a search engine prefer this so-called *relevance ranking*? and why did Morik et al. [2020] call this ranking a ranking by the "true average relevance"? I will discuss this below.

Ranking Algorithm	RR	AP	nDCG
relevance ranking (unfair)	0.70	0.70	0.76
interleaved ranking (fair)	0.83	0.72	0.82

Table 2: Evaluation results for Cooper's example (Robertson [1977])

4 Discussion

To understand the origins of the "true average relevance" ranking, we have to dig a bit deeper into Robertson's probability ranking principle. The principle states that under certain conditions, a ranking by the probability of relevance as done by Morik et al. [2020] will produce the best overall effectiveness that is obtainable on the basis of the data. Those conditions are the following:

1. The relevance of a document to a request does not depend on other documents in the collection;
2. The principle relates only to a single request;
3. Relevance is a dichotomous variable.

Condition 1 is clearly violated in our examples. In the example with right-leaning and left-leaning users, knowing that a user likes one right-leaning document should drastically change the probability of relevance for the other documents. Condition 3 is violated if we use graded relevance and evaluation measures like (n)DCG. If our aim is to build a fair ranker, then we cannot blindly apply the probability ranking principle.[2]

5 Conclusion

Unfair rankings were discussed already 45 years ago by Cooper and Robertson, even though they did not used the term "fairness" as such. If the conditions for the probability ranking principle are not met, then we *a)* may not get the overall best quality ranking; and *b)* instead get a biased ranking that systematically and *unfairly* favours the majority group of users over the minority group.

Sadly, what happened to Latanya Sweeney may have been the following: Google optimized its advertisement ranker using a click-based relevance estimator that ranks advertisements by their probability of relevance under the conditions of the probability of ranking principle.[3] These conditions are not met for the query "Latanya Sweeney". There are at least two groups of people: *1)* A majority group that clicks background checks for "black names", and *2)* A minority group that clicks for instance on advertisements for connecting on social media. Even though both groups may be roughly equal in size, Google only showed the top advertisements of the majority

[2] While I think Morik et al. [2020] are worthy recipients of the SIGIR 2020 Best Paper Award, best papers also deserve extra scrutiny: The paper cites Robertson [1977] without checking the conditions of the probability ranking principle as follows: *"Fortunately, it is easy to show (Robertson 1977) that sorting-based policies* $\pi(x) = \textit{argsort}_{d \in R(d|x)}$ *are optimal for virtually all* [*evaluation measures*] *commonly used in IR (e.g. DCG)."*

[3] Google is evil is another explanation.

group. Google thereby showed biased, racist results that adversely impact the minority group. Furthermore, the ranking probably did not even optimize for advertisement revenue, because the preconditions for the probability ranking principle were not met.

The most important message here: The relevance of the results of a search algorithm (and therefore the search engine's revenue) is not necessarily at odds with the fairness of the results. Robertson and Cooper's example shows that there are cases where improving the quality of the results (measured in RR, AP or nDCG) also improves the fairness of the results.[4]

References

Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 429–438, 2020. (Awarded Best paper in 2020).

Cathy O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown, 2016.

Stephen Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33(4): 294–304, 1977.

Latanya Sweeney. Discrimination in online ad delivery. *Communications of the ACM*, 56:44–54, 2013. (arXiv:1301.6822).

[4] Note that to get a truly fair ranking, we should frequently switch both groups when interleaving the documents, starting with the minority group with a probability proportional to the size of the group. This will somewhat negatively impact the expected search quality.