Bias-Mitigating News Search with BiasRank

Tim Menzner[1][a] and Jochen L. Leidner[1,2][b]
[1]*Information Access Research Group, Center for Responsible Artificial Intelligence Research (CRAI), Coburg University of Applied Sciences, Coburg, Germany*
[2]*Department of Computer Science, University of Sheffield, Sheffield, U.K.*

Keywords: News Bias Detection, Information Retrieval, Bias-Aware Ranking, Large Language Models, News Recommendation Systems, Search Result Fairness, Re-Ranking.

Abstract: As geopolitical adversaries as well as internal commercial and political actors target democracies with disinformation campaigns, it is increasingly necessary to filter out biased reporting. Some automatic success has recently been achieved in this task. For further progress, web search engines need to implement news bias resistance mechanisms for ranking news stories. To this end, we present BiasRank, a new approach that demotes articles exhibiting news media bias by combining a large neural language model for news bias classification with a heuristic re-ranker. Our experiments, based on artificially polluting a (mostly neutral) standard news corpus with various degrees of biased news stories (biased to varying extents), inspired by earlier work on answer injection, demonstrate the effectiveness of the approach. Our evaluation shows that the method radically reduces news bias at a negligible cost in terms of relevance. In turn, we also provide new metrics for the evaluation of similar systems that aim to balance two variables (like relevancy and bias in our case). Additionally, we release our test collection on git to support further research on de-biasing news search.

1 INTRODUCTION

Web search engines, such as Google, Baidu, Qwant, Yandex, DuckDuckGo and others, as well as news recommender engines, such as Google News, are powerful tools for seeking specific information as well as for getting news stories. However, it has been shown that these systems suffer from various types of bias (Gharahighehi et al., 2021; Wendelin et al., 2017), and given the pervasiveness of Web search in our lives, there is a looming threat of manipulating online audiences for political or monetary gain. To help counter this issue, in this paper we explore approaches for reducing media bias in the ranking of news stories. Specifically, we address the following research question:

Research Question (RQ): *How can we achieve less biased rankings in a news search or news recommendation context?*

The main contributions of this work are as follows:

- We describe BiasRank, a new hybrid method for ranking news stories, promoting objective news reports and demoting individual stories and websites that suffer from media bias;
- We propose a dynamic method to update the index with LLM-generated information only when a document is requested, minimizing the need for frequent and costly LLM calls;
- We outline a set of metrics to measure bias in query results and its (or any other metric's) trade-off with result relevance;
- We present an empirical evaluation that demonstrates the efficacy of BiasRank on a news corpus;
- We release a demo of our system, as well as our test collection, to the public in order to foster more discussion and encourage future work;

To analyze ranking relevance and bias *together*, and to explore the trade-offs doing so, we need five ingredients: 1. a data collection (we combine a news corpus with injected known-bias stories), 2. a set of queries (we created a set), 3. a set of relevance judgments (QRELs, we created judgments for two retrieval methods for the top-40 for our topics), 4. a set of bias assignments for retrieved documents (we use

[a] https://orcid.org/0009-0005-9753-9364
[b] https://orcid.org/0000-0002-1219-4696

Menzner, T. and Leidner, J. L.
Bias-Mitigating News Search with BiasRank.
DOI: 10.5220/0013755200004000

In *Proceedings of the 17th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2025) - Volume 1: KDIR*, pages 436-447

both a lexicon-based baseline and a state-of-the-art, custom-fine tuned, 27-class news bias neural transformer model) and 5. an evaluation metric that combines relevance and bias (a new one is proposed below).

Note that we address a different problem from (Joachims, 2002) or (Craswell et al., 2008), who both address *positional bias* irrespective of the content of documents at particular positions, whereaswe address *news bias*, i.e. the neutrality (pr not) of documents together with their ranks. We are not aware of any prior work on true news bias re-ranking, but (Ye and Skiena, 2019), who approximate news bias analysis with sentiment analysis, comes closest among prior approaches.

Figure 1: Search results for a query in BiasRank before (top) and after (bottom left and bottom right) re-ranking based on two different settings.

2 RELATED WORK

Bias, propaganda and disinformation in media have been widely studied (Herman and Chomsky, 1988; Lippmann, 1922). Building on this foundational work, recent studies have empirically established the presence of bias in both Web search and social media (Bakshy et al., 2015; Gezici et al., 2021). In the following sections, we review work on various bias types and related fields relevant to news search.

2.1 Media Content Bias

Some biases may stem directly from the agenda of an owner or other decision makers, others might be artifacts of the way systems have been built and how models have been trained. Increasingly there is also "customer orientation" or "bias by demand", i.e. journalists write about things news consumers care about and click on based on their biases, which is its own selection bias, or even clickbait (Wendelin et al., 2017).

Lauw, Lim and Wang (Lauw et al., 2006) argue that bias and controversy are connected, and should therefore be analysed together: the same degree of bias observation at the surface may count for more if observed for a less controversial topic.

Fine-grained models and systems for the detection of propaganda (Da San Martino et al., 2019) and media bias (Menzner and Leidner, 2024b; Menzner and Leidner, 2024c) have been proposed, based on machine learning methods, where propaganda denotes voluntary influencing for political gain whereas media bias is a broader concept that includes propaganda and also involuntary distortions. These fine-rained detection methods inspire our choice of a neural classifier in BiasRank, as they open up the way for a reliable, automated detection of document bias based on its actual content.

2.2 Gender Bias in Text & Search

Gender bias in text and in search has received substantial attention in its own right (e.g., (Costa-Jussà, 2019)). This line of work is partly motivated by gender stereotypes that emerge through machine learning when translating between languages. In some languages, the grammatical gender reflects the biological sex of the person holding a profession, whereas in others it does not. For example, nurse in English is gender-neutral, while Krankenschwester in German refers only to female nurses. Ratz, Schedl and Kopeinik (Ratz et al., 2024) look at gender bias and evaluate their on bias metric for it against past work on a recent collection of bias-sensitive topics and documents from MS MARCO data.

2.3 Political Bias on the Web & Social Media

(Kulshrestha et al., 2019) propose a framework to quantify political bias in social media search results by disentangling bias introduced by input data from that introduced by the ranking system, and, through empirical analysis of Twitter queries during the 2016 US presidential primaries, they find that both

sources significantly shape the political bias observed in search results.

A study by Epstein and Robertson (Epstein and Robertson, 2015) investigated what they called "the search engine manipulation effect", finding that biased search rankings can indeed shift the voting preferences of undecided voters.

2.4 Bias in Rankings

(Gharahighehi et al., 2021) address the problem of popularity bias ("rich get richer") in rankings.

(Ovaisi et al., 2020) consider position bias and selection bias in rankings in recommender engines that uses learning to rank. The authors adapt a bias correction method from the older statistical literature to the recommendation ranking scenario and demonstrate superior accuracy compared to unbiased rankings. Crucially, their method does *not* inspect the actual documents at each rank.[1]

Fairness using protected Attribute Labels has been the topic of the *Fair Ranking Track* shared task at US NIST's *Text REtrieval Conferences* (TREC) (Ekstrand et al., 2022), which targeted fair exposure of individual attributes or groups of them, based on Wikipedia documents. Raj and Ekstrand compared different evaluation metrics for fair ranking and found the Attention-Weighted Rank Fairness (AWRF) (Sapiezynski et al., 2019; Raj and Ekstrand, 2022) to be the most generally useful metric for single rankings with its adaptability to different models, target distributions, and difference functions (Raj and Ekstrand, 2022).

In the FAIR Ranking Track, the product of AWRF and nDCG (Järvelin and Kekäläinen, 2002) is formed to give relevance and fairness the same weight in the evaluation of sub-task 1 at that shared task. citeDaietal:2024:KDD provide a survey of the various challenges around bias in IR. *Note that the extensive body of work on statistical distortions in search results (biased rankings) is different from the topic of this paper (biased-language news rankings).*

2.5 Previous Attempted Remedies

(Jaenich et al., 2024) describe adaptive re-ranking methods aimed to increase the visibility of relevant but underrepresented groups in the re-ranking phase of a two-stage retrieval process comprising document ranking and re-ranking.

. In the context of news recommendation, the technical report (Wu et al., 2022) describe a fairness-aware ranking approach that models users' interest via user embeddings, obtained via adversarial learning also from click data.

In contrast to these models, our heuristic approach is not only simpler, it can also be implemented in settings where click data is unavailable.

(Park et al., 2012) present NewsCube, an aspect-oriented news browser prototype; by presenting multiple aspects of each news story they aim to mitigate news bias.[2] The advantage of this approach is that it avoids automatic censorship, intentional or otherwise. But the approach implies that users are actually interested in investigating a broad range of alternative viewpoints.

(Hu et al., 2019) study political partisanship bias in the snippets of the Google Web search's SERP using a lexicon approach.

Different design choices for bias-aware web searches were investigated by (Paramita et al., 2022). Even though their prototype was a mock-up that did not actually assess document bias, their findings confirm the utility of a re-ranking approach for such systems.

Perhaps closest in spirit to our approach is the work of Ye and Skiena (Ye and Skiena, 2019), who describe *MediaRank*, a method and Website that ranks >50,000 media Websites based on the factors peer reputation (where number of citations is taken to be a proxy for reputation), reporting bias/breadth (where sentiment differences of a large set of left-wing and right-wing individuals towards them is used as a proxy), bottom-line financial pressure (using bot and ad activity as a proxy) and popularity (using Alexa rank as a proxy). Unlike these statistical corrections , we directly analyze content to detect linguistic bias in news items; our system also focuses on news bias, which is estimated directly by a custom model rather than by using a proxy.

3 METHOD

Our goal is to take into account the content bias of all individual documents in a collection, but in a way that only minimally impacts the typical IR indexing and retrieval pipeline. We also aim to facilitate implementation of our method as part of existing legacy

[1] Re-ordering the items in a ranking because item positions may have incurred a bias *as per their position alone* is different from inspecting the textual *content* of each item and estimating a *content bias score*, which is what we propose here.

[2] At the time of writing, the system is no longer available on the Internet.

indexing and retrieval pipelines that may be hard to change, but which we wish to enrich with our method for adding resilience in the face of news bias (Figure 2).

We propose a heuristic search function that combines a relevance model and an anti-bias model to calculate a ranking score for a document d and a query q, based on a score $rel_{relevance}$ indicating the relevance for q, as well as a score $bias_{document}$ indicating how biased the content of a document is, through simple linear interpolation as follows:

$$\begin{aligned} BiasRank(q,d) = &(1-\lambda)\cdot rel_{document}(q,d) \\ &+\lambda\cdot(1-bias_{document}(d)) \end{aligned} \quad (1)$$

where the linear interpolation weight λ controls the degree of the influence of the bias score in the overall score. $0 \leq \lambda \leq 1$, where $\lambda = 0$ means that the bias-based re-ranking will be switched off, and BiasRank behaves like a pure relevance ranker, whereas $\lambda = 1$ means that the relevance ranking term disappears, so BiasRank decays to perform a search for the least biased documents, not taking any relevance into account at all.

To assess a document's bias, we use BiasScanner (Menzner and Leidner, 2025; Menzner and Leidner, 2024a; Menzner and Leidner, 2024c), a large custom language model (LLM) fine-tuned with training data comprising biased sentences from news articles, annotated with bias type and intensity, to identify all biased sentences and determine the intensity (bias strength on a scale of 0 to 1) of the bias in each individual, biased sentence. For this experiment, we opted for the GPT-3.5 variant of BiasScanner. Let a document have N sentences in total, of which n are classified as biased with a respective intensity b. We can then use this information to calculate the overall bias of a document: *pervasiveness* as the proportion of biased sentences and *strength* as the mean bias intensity of these sentences. By combining these two measures, we obtain a single overall bias score $bias_{document}$, which reflects both the extent and intensity of bias within the document:

$$\begin{aligned} \text{pervasiveness} &= \frac{n}{N}, \quad \text{strength} = \frac{1}{n}\sum_{i=1}^{n} b_i \\ bias_{document} &= \frac{\text{pervasiveness} + \text{strength}}{2} \end{aligned} \quad (2)$$

4 IMPLEMENTATION

We implemented our score as a re-ranking procedure on top of length-normalized TFIDF and BM25 scores

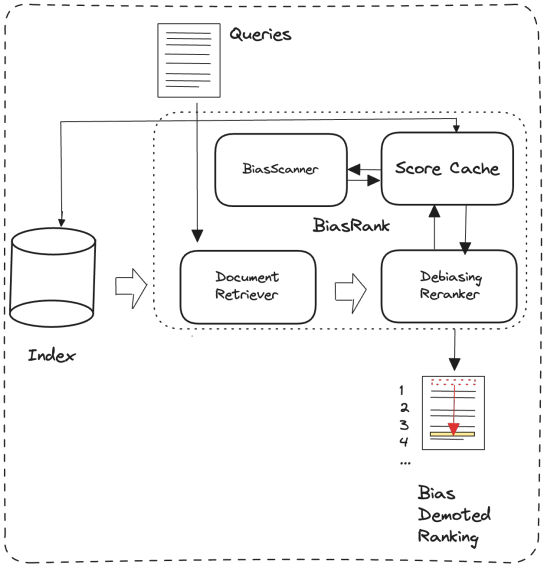

Figure 2: BiasRank Architectural Overview.

as provided by Apache Lucene search library (Andrzej Białecki, 2012), to compare performance across standard relevance models. Our system is based on the PyLucene (9.7.0) [3] wrapper. The architecture of the system is shown in 2.

1. When a query retrieves a document from the index, the de-biasing ranker first checks whether the document already includes the field we use to specify its bias information.
2. If this field is absent, the ranker then determines if the document's bias has already been cached in a Redis(Sanfilippo, 2009) database.
3. If the bias is not cached, the document is forwarded to a component that evaluates its bias. By default, we employ the BiasScanner model for this assessment, though it can be easily replaced with any comparable method, as long as it returns a score between 0 and 1 for each document.
4. Once the bias score is generated, it is stored in the cache, and the bias field is added to the document in the index.

By only rating documents that actually appear in our queries and storing the results, we optimize our system's efficiency and reduce unnecessary computation as well as cost and energy consumption, ensuring that subsequent queries can retrieve bias information quickly and accurately without redundant processing.

We re-scale all Lucene relevance scores to $[0;1]$ using the minimum and maximum returned lucene

[3] https://lucene.apache.org/pylucene/

score for all n results of the query with a linear normalization; our bias score assigned to a document is always between 0 and 1 by definition (In practice, the lower bound of 0 can occur somewhat frequently. However, the upper bound of 1 is rarely reached because it would require a document made up entirely of biased sentences, without any generic filler text, and each sentence would need to be extremely biased.).

5 EVALUATION

5.1 Retrieval Setup

Our evaluation protocol is inspired by *answer injection* (Leidner and Callison-Burch, 2003), a method to evaluate question answering systems by planting known answers in large background corpora in a way so as to remember where (in which document ID) the correct answer to any one particular question was to be found. Following this protocol, we first create an artificially polluted corpus from a assumed-neutral background corpus by planting news stories known to us to be biased inside the background corpus of news, which can be expected to be mostly unbiased; we call this enriched corpus the "polluted corpus".

5.2 Collection

We utilize the Reuters TRC2 English sub-corpus as our background corpus: for our experiments, we assume the great majority of Reuters stories to be unbiased [4] and "pollute" it with news stories known to be biased. While the notion of a completely unbiased news agency is likely an unattainable standard, not least because individual definitions of bias may vary depending on perspective, Reuters is often considered one of the news agencies that come closest to achieving this goal (Ad Fontes Media (eds.), 2024; Budak et al., 2016).

The TRC2 collection is a collection of news reports from the Reuters news agency (owned by the Thomson Reuters Corporation and distributed by US NIST for research purposes) for English. TRC2 was designed originally to be time-aligned with another corpus covering blogs (BLOG09), so it contains news from 14 months starting with the year 2009.

The biased news stories injected in the corpus were manually collected by searching for biased articles on the Fox News website that addressed topics reported in TRC2 during the given time period. Fox News was selected as a source due to its convenient search function, which allows for easy access to articles from the relevant time-frame and keywords. Additionally, its well-documented right-wing bias (Martin and Yurukoglu, 2017; Bernhardt et al., 2020) facilitates the identification of articles that exhibit bias while covering pertinent topics.

The limitation of gathering biased articles from only one side of the bias spectrum does not impede our experiment, as the bias score calculation we rely on is agnostic to the direction of bias, whether right-wing, left-wing, or otherwise. As long as a document is biased, it is likely to yield a high bias score.

Besides time-frame, the specific topics where also chosen based on the likely contentiousness of the event, because we want to have a realistic likelihood that biased (as well as unbiased) stories about these topics are retrieved; so if the topic is not somewhat controversial, there may be not enough data in the intersection set between relevant and biased stories.

Overall, 85 biased articles covering 17 different topics including "same-sex marriage", the "auto bailout" and "Obama's presidential campaign and victory" were picked and injected.

5.3 Queries

We constructed set of 40 queries (often called "topics" in IR) based on the 17 different topics identified for Section 5.2. We ensured that the queries themselves were not inherently biased, avoiding, the explicit request for a Fox News article or the use of loaded terms. Instead, we formulated queries that one might use when genuinely seeking information about a topic without pre-existing bias (e.g., "obama stimulus package" rather than "obama socialist stimulus fox news").

5.4 Relevance Judgments

The two co-authors annotated a set of documents for relevance with respect to the 40 queries. Given top-k retrieval with $k = 40$, there are less than $40 \times 40 \times$ IR methods=2 $= 3,200$ QRELs to produce, however in practice there is substantial overlap between documents retrieved by the vector space model with TFIDF weighting and the binary probabilistic model with BM25 weighting. We divided the data in three groups, one per annotator for single annotation and a smaller partition with $N = 100$ doubly annotated records to be able to determine inter-annotator agreement. We annotated the JSON representation of the QREL tuples that included the question and the title as well as the first 512 characters of the document

[4] We counted 645 opinion pieces in the TRC2 dataset among 1,312,775 documents ($< 0.05\%$).

directly in a text editor. Each document was categorized as either "relevant" or "not relevant". For a $N = 8$ query sample and $k = 40$ top-k retrieval results, we constructed QRELs with two raters; the resulting inter-annotator agreement observed was 95.55% (raw overlap) and 0.91 (in terms of Cohen κ), which can be described as nearly perfect agreement, bolstering confidence in the quality of annotations.

5.5 Baseline

To enable the assessment of the impact of the news bias model, we also implemented a simple lexicon baseline method that works as follows: a small set of terms are looked up from a hashtable and each sentence with at least one match encountered while going through a news story increases a counter. The overall bias score of an article is then calculated by dividing this counter with the total number of sentences. `Biaslex-baseline-IPM21` uses the list of 76 English bias indicator terms from Spinde et al. (Spinde et al., 2021) whereas `Biaslex-baseline-KDIR2025` uses our own list of 48 bias terms made up from introspection and browsing the Web for resources explaining for human readers how to identify news biases, as well as term obtained by prompting ChatGPT-4o to output the 50 terms most strongly indicative of news bias.[5]

5.6 Evaluation Metrics

We evaluate several retrieval methods against our relevance judgments before and after re-ranking with our method. To assess the overall bias in a set of n search results for a given query, we calculate the sum of the bias scores assigned to each document, applying a logarithmic weighting based on its position in the ranking. This approach gives greater weight to higher-ranking documents, ensuring they have a larger influence on the overall bias, as we consider the top results to be the most significant in shaping the overall perception of the query.

$$\text{bias}_{\text{results}} = \sum_{i=1}^{n} \frac{\text{bias}_{\text{document}}}{\log_2(i+1)} \tag{3}$$

To measure relevance of a set, we decided to use Normalized Discounted Cumulative Gain (NDCG) provided by `trec_eval` (Järvelin and Kekäläinen, 2002) because it accounts for document position in a manner similar to our bias calculation, thereby enhancing comparability. Since NDCG also relies on normalization with the maximum possible DCG, we normalized $\text{bias}_{\text{results}}$ using the maximum possible bias of the given set ($\text{bias}_{\text{results}}$ when sorted in descending order of bias). However, our overarching interest is whether bias is reduced in a way that does not, or not substantially, affect relevancy in a negative way. To this end, we can define a combined metric, the Linear Re-ranking Impact Score (LRIS), based on the delta of $\text{bias}_{\text{results}}$ and $\text{relevancy}_{\text{results}}$ in percent before and after the re-reanking:

$$\text{LRIS} = -1 \times \Delta\text{bias}_{\text{results}} + \Delta\text{relevance}_{\text{results}} \tag{4}$$

When the decrease in bias after re-ranking is larger than the decrease in relevancy, the RIS will be positive. If relevancy decreases larger than bias, it will be negative. Besides this linear trade-off metric, we also calculate the delta in an Adapted Harmonic Mean (AHM) between bias and relevancy of the set before and after re-ranking (similar like a F-score combines Precision and Recall). This Non-Linear Re-ranking Impact Score (NRIS) is more sensitive to small (absolute) improvements when relevance or bias values are low.

We conducted a second evaluation focusing solely on the top-k results out of our n, without applying position-based weighting within this window. In this case, relevance and bias scores for the k out of n results may change due to back-filling, as documents in the top-k can be replaced by others with different relevance or bias levels trough the re-ranking. To remove position-based weighting, we replaced NDCG with Precision for calculating $\text{relevance}_{\text{top-k}}$. To ensure comparability, $\text{bias}_{\text{top-k}}$ was also calculated in a precision-like manner in this round, representing the proportion of documents in the $top-k$ with a bias score greater than zero. We chose a k of 10, as this is also the standard number of results you would get on the first page of many search engines.

LRIS and NRIS rely on effective bias-scoring methods. A system that assigns high bias scores to unbiased documents may still perform well by demoting these misclassified documents, while failing to address any true bias it cannot measure. To address this, we use a second version of LRIS, the Injection-based Linear Re-ranking Impact Score (ILRIS). Under our premise that the injected documents are biased and the TRC2 documents are neutral, we assign bias values of 0 and 1 accordingly. We then calculate the ILRIS like we would LRIS to assess the effects of the re-ranking, which is still done using the scores of the

[5] both lexica as well as the URLs of the injected articles and all queries with the corresponding qrels can be found *here* online (https://github.com/Timperator2/BiasRankReproducibility)

respective bias-scoring method.

$$\text{AHM} = \frac{2 \cdot \text{relevance}_{\text{results}} \cdot (1 - \text{bias}_{\text{results}})}{\text{relevance}_{\text{results}} + (1 - \text{bias}_{\text{results}})} \quad (5)$$
$$\text{NRIS} = \Delta\text{AHM}$$

5.7 Results

5.7.1 Bias-Scoring Methods

Table 1 compares the BiasScanner method for determining the bias of news articles with the two baselines described in Section 5.5 and a third baseline in which bias values are assigned randomly as numbers between 0 and 1. For each retrieval method (BM25 and TFIDF), the same 40 queries with 40 hits were used, resulting in 1257 unique documents for BM25 and 1229 unique documents for TFIDF, respectively.

To make the scores assigned by each method more comparable with one another, bias scores are normalized using the lowest and highest assigned scores for each individual query as bounds.

All methods except for the random baseline assign significantly higher bias values to the injected documents compared to the TRC2 documents. This indicates that the methods align with our premise that TRC2 documents can generally be considered unbiased by default, while injected documents are biased.The same applies when examining the average ranking of TRC2 and injected documents among the top-40 retrieved documents for each query, after sorting them in descending order by bias.

Across all methods except the random baseline, the injected documents consistently rank higher (indicating greater bias) than the TRC2 documents, with this difference being greatest with BiasScanner. BiasScanner also performed best in terms of F1. Because a simple threshold approach, in which a document's bias score had to exceed a certain value, was not feasible due to differences in scoring methods across the compared approaches, the confusion matrix for calculating the F1-score was derived using an alternative approach to ensure comparability: based on our premise, for a query with n injected documents in its results, the n strongest-biased documents should be the injected ones. Therefore, true positives are injected documents among the n most biased, false positives are TRC2 documents in the n most biased, true negatives are TRC2 documents not in the n most biased, and false negatives are injected documents not in the n most biased.

Even-though all methods perform way better than random on this metric, overall F1 is still rather low (between 0.237 for Biaslex-baseline-KDIR2025 and 0.339 for BiasScanner) due to a relatively high number of false positives. This has two reasons.

First of all, while our premise generally holds true, it is, of course, an oversimplification. Even if most Reuters articles are unbiased, the sheer overrepresentation of these articles in the dataset (for all unique documents retrieved with BM25, 1,177 are from TRC2, while only 52 belong to the injected ones, with similar proportions for TFIDF) ensures that a low percentage of biased articles can lead to a high number of false positives in our setup.

Secondly, the systems themselves are imperfect, as evidenced by examples where relatively high bias scores were assigned to neutral-looking Reuters articles. In addition to formatting issues such as some news reports missing proper punctuation (which can disrupt the calculation of the overall bias score, partly based on the percentage of biased sentences), quotes containing biased content are also an important aspect that can lead to bias being detected in otherwise neutral articles. These phenomena and their impact is discussed in more detail in Section 7.

Interestingly, even though IPM21 mainly contains words associated with topics that are often associated with bias rather than words that directly indicate biased language, it still performs relatively well. Overall, BiasScanner generally achieves the best performance in detecting bias for all tested methods, its good performance is consistent with other, independent evaluation on datasets specifically constructed for bias detection (Menzner and Leidner, 2024c).

5.7.2 Re-Ranking

Table 2 provides a comparison of the averages of relevance and bias metrics after re-ranking using BiasScanner with varying bias weightings (λ) for full set ($n = 40$) and $top-k = 10$ retrieval across 40 queries using BM25 and TFIDF.

As expected, increasing bias weight reduces bias post re-ranking but also decreases relevancy. The table shows that optimal balance is generally achieved with higher weightings, though peak LRIS, NRIS and ILRIS values typically occur between 0.5 and 0.75. The highest LRIS for n is 0.181 (TFIDF, $\lambda = 0.62$) and 0.658 (BM25, $\lambda = 0.74$) for top-k. The NRIS peaks at 0.162 (BM25, $\lambda = 0.72$) for n and 0.147 (BM25, $\lambda = 0.59$) for top-k. IRLIS is at its highest for n at 0.228 (BM25, $\lambda = 0.68$) and 0.399 (BM25, $\lambda = 0.70$).

LRIS scores are way higher for $top-k$ than for n, while NRIS differences are less pronounced and peak earlier for $top-k$. LRIS uses relative percentage changes, weighting small bias reductions similarly to larger relevance reductions.

Table 1: Evaluation of different bias-scoring methods including the average assigned bias score and the average place when ranked by bias for TRC2 with injected documents (normalized for better comparability). The results confirm the suitability of BiasScanner as a method for assessing document bias in this scenario.

Method	TRC	INJ	TRC-Rank	INJ-Rank	F1-Score
Random baseline	0.504	0.512	20.48	20.13	0.048
Biaslex-baseline-IPM21	0.063	0.138	20.94	12.86	0.310
Biaslex-baseline-KDIR2025	0.056	0.188	21.08	10.79	0.237
BiasScanner	0.296	0.772	**21.21**	**7.10**	**0.339**

Table 2: Comparison of averages of relevance and bias metrics after re-ranking with varying bias weightings (λ) for full set ($n = 40$) and $top-k = 10$ retrieval across 40 queries using BM25 and TFIDF. The table includes changes in relevance (ΔR), bias as rated by BiasScanner (ΔB) and bias measured via injected documents (ΔB_I), LRIS, NRIS and ILRIS metrics (all as defined in 5.6), as well as the percentage of queries with an improvement in LRIS and NRIS. ΔTRC and ΔINJ show the average change in ranking of TRC2 and injected documents, with top-k only looking at the top-10. The results show that the general principle works, providing an overview of which parameters correspond to the expected trade-off between loss of relevancy and gain in neutrality, and indicate that the sweet spot lies somewhere between a bias weighting of 0.5 and 0.75.

Setup	ΔR	ΔB	LRIS	NRIS	$\uparrow LRIS$	$\uparrow NRIS$	ΔTRC	ΔINJ	ΔB_I	ILRIS
BM25$_n$										
$\lambda = 0.25$	-2.5%	-9.5%	0.070	0.069	90.0%	**97.5%**	0.2	-4.1	-10.2%	0.077
$\lambda = 0.5$	-6.7%	-20.3%	**0.136**	0.138	**92.5%**	**97.5%**	0.6	-10.3	-24.8%	0.182
$\lambda = 0.75$	-16.2%	-29.8%	**0.136**	**0.160**	82.5%	95.0%	1.0	-17.8	-38.3%	**0.221**
BM25$_k$										
$\lambda = 0.25$	-3.59%	-29.35%	0.258	0.078	70.0%	67.5%	-0.7	-3.2	-11.7%	0.081
$\lambda = 0.5$	-13.3%	-65.6%	0.523	**0.141**	**87.5%**	**75.0%**	-3.2	-12.0	-41.3%	0.280
$\lambda = 0.75$	-34.4%	-100.0%	**0.656**	0.100	82.5%	65.0%	-7.1	-23.7	-72.5%	**0.381**
TFIDF$_n$										
$\lambda = 0.25$	-0.7%	-10.2%	0.095	0.066	**100%**	**92.5%**	0.2	-4.4	-7.2%	0.066
$\lambda = 0.5$	-5.9%	-21.7%	0.158	0.119	92.5%	90.0%	0.6	-10.9	-24.2 %	0.182
$\lambda = 0.75$	-12.9%	-30.6%	**0.177**	**0.138**	87.5%	87.5%	0.9	-17.8	-33.8%	**0.209**
TFIDF$_k$										
$\lambda = 0.25$	-9.4%	-28.3%	0.189	0.039	60.0%	52.5%	-0.8	-3.1	-6.3%	-0.032
$\lambda = 0.5$	-16.1%	-63.5%	0.474	**0.070**	**87.5%**	**72.5%**	-3.0	-11.8	-37.5%	0.213
$\lambda = 0.75$	-29.6%	-94.4%	**0.648**	0.043	82.5%	57.5%	-6.8	-26.1	-65.0%	**0.354**

In contrast, NRIS focuses on absolute values, which makes it less affected by high percentage changes in small values, despite small values having a greater effect on the Adapted Harmonic Mean. Thus, in cases with low bias where relevance is crucial, NRIS may be a better metric for overall performance evaluation.

Although LRIS and NRIS improved for most queries across all parameters, there was at least one query in all but one case where the relevancy-bias ratio either did not improve or worsened. While ILRIS also shows improvement in most cases, there is one setup where re-ranking actually results in a slightly worse trade-off between bias and relevance, according to this metric.

Overall, the ILRIS scores confirm that the re-ranking indeed operates in line with our initial premise when using BiasScanner, as they strongly correlate with the LRIS scores according to Pearson correlation coefficient, r(10) = .768, p = .0035. The correlation between bias reduction in percent using BiasScanner values (ΔB) and bias reduction in percent based on the demotion of injected documents (ΔB_I) is even stronger, r(10) = 0.907, p < 0.0001.

Interestingly, the weightings that show the highest percentage of improvements in LRIS and NRIS are often not the same as those associated with the highest average scores in these metrics. This suggests a trade-off: one can opt for smaller yet more consistent improvements or pursue the potential for larger gains, which also carries the risk of negative outcomes.

Generally speaking, a medium-high value of λ tends to be optimal. For the or the $top-k$ selection, where the differences a especially high, lower

values may fail to adequately filter out biased documents, while excessively high values can lead to diminishing returns in bias reduction for many queries (while the improvement on others is still high enough to drive up the total average).

As an additional insight, de-biasing can occasionally even improve relevancy by allowing more relevant, unbiased documents to replace non-relevant, biased ones. This occurs for at least 5% of queries ($top-k$ TFIDF with $\lambda = 0.25$) and up to 30% (n TFIDF with $\lambda = 0.5$), averaging around 16%. Consequently, this drives up LRIS and NRIS scores, as de-biasing consistently reduces bias.

5.7.3 Re-Ranking with Different Bias-Scoring Methods

Table 3 shows differences in system performance when employing different bias-scoring methods apart from BiasScanner. The system with BiasScanner outperforms the other variants in all metrics and shows the clearest correlation between between bias reduction using the automatically assigned values and bias reduction based on the demotion of the injected documents. As described in 5.6, the meaningfulness of LRIS depends in part on the effectiveness of the bias-scoring methods in accurately identifying bias. (In line with our premise, a higher value of $r(\Delta B, \Delta BI)$ indicates greater meaningfulness). Consequently, LRIS may be more effective for comparisons within a single method rather than between different methods. Still, even for ILRIS, IPM21 and KDIR2025 scores remain relatively low.

6 DEMO

As we believe that actually interacting with a system makes it easier to understand what it is about than mere walls of text and tables, we also implemented a live demo accessible under https://biasscanner.org/BiasRankWebDemo

A screenshot from the demo is shown in 1

For performance reasons, the web version of our demo prototype currently does not support actual live search. Instead, we have pre-cached the search results for the 40 queries (see 5.3) using the BM25 algorithm in Lucene. When a query is entered, the system retrieves the results of the most similar cached query, with similarity determined by a combination of semantic matching using word2vec(Mikolov et al., 2013) embeddings and cosine similarity, in combination with exact string matching done via Levenshtein distance. Users can adjust the search ranking by using a slider going from 0 to 1 in steps of 0.01, which allows them to control the extent to which bias influences the ranking. This interactive demo illustrates how the trade-offs between bias and relevance, as quantified in our evaluation, manifest in a practical setting.

7 LIMITATIONS

Obviously, the quality of our re-ranking approach is highly dependent on the accuracy of the system used to assess document bias in the first place (as demonstrated by comparisons with the baseline word lexica in 5.7.1 and 5.7.3). Additionally, the performance of the ranking algorithm used for determining relevance, is just as crucial.

That said, we like to emphasize that our main contribution lies not in the specific bias assessment system, but in providing a general framework that can be applied across such systems.

We are aware that the number of biased documents is relatively low, at least compared to other works. However, we believe that this number is sufficient to demonstrate the applicability of our method as the observed effects were strong and the correlations described in 5.7statistically significant. Overall, comprehensive bias analysis of every document is an expensive operation, more so than other typical IR text analysis task (e.g. spam filtering, topic classification). Content bias analysis must be carried out in full, as processing just the beginning of a document could lead to gaming the method.

One question is whether a “mostly neutral stories retrieved” setup is actually desirable at all: it could be that a more diverse, but balanced mix of neutral news stories as well as news stories with various biases is more helpful, depending on the motivation of the news search. We content that search engines should make such choices transparent to the end user, although it is known that most users never modify defaults.

The influence of quotes on the bias of an article is a topic worthy of its own debate. Currently, our system does not differentiate between quotes and non-quotes. One could argue that simply reproducing a biased statement made by someone as part of an otherwise impartial report should not increase the article’s bias score. However, when a publication selectively chooses whom and what to quote to advance a particular narrative, quotes can become tools of media bias. Ultimately, the impact of a quote depends on its overall context and the role it plays within the article.

Finally, from an ethics perspective, the decision

Table 3: Comparison of system performance using different bias-scoring methods for the full set ($n = 40$) and $top-k = 10$ retrieval across 40 queries. LRIS and ILRIS values are averages of both BM25 and TFIDF for bias weightings $\lambda = 0.25$, $\lambda = 0.5$, and $\lambda = 0.75$. Pearson correlation is between bias reduction using the bias values returned by the method and bias reduction based on the demotion of injected documents. The results show the clearest correlation between between bias reduction using the automatically assigned values and bias reduction based on the demotion of the injected documents when using BiasScanner.

Method	$\mathbf{LRIS}_n$	$\mathbf{LRIS}_k$	$\mathbf{ILRIS}_n$	$\mathbf{ILRIS}_k$	$\mathbf{r}(\Delta B, \Delta BI)$
Random baseline	0.0825	-0.097	-0.005	0.039	-0.361
Biaslex-baseline-IPM21	0.058	0.067	0.001	-0.017	0.740
Biaslex-baseline-KDIR2025	0.085	0.148	0.014	0.022	0.876
BiasScanner	**0.129**	**0.458**	**0.156**	**0.213**	**0.907**

which sentences are biased is a sensitive one; users may argue it should not be up to a technology provider to decide what is biased; however, we consider this question is not much different from leaving the relevancy ranking to a third party. What might be presented as relevant to a user might already be the result of bias in the process of relevancy calculation, especially when the algorithm considers a personal profile for making its selection. We mitigate user acceptance risk by selecting a bias model that generates textual explanations for each sentence classified as biased.

8 SUMMARY, CONCLUSION AND FUTURE WORK

We presented BiasRank, the first heuristic re-ranking method that is informed by bias as well as relevance: With "bias" refering to a full news content bias analysis carried out on the sentence-level for each indexed document (content bias profiling). Our method can equally be used for recommendation and search as step added on top of the initial ranking. We further provide appropriate metrics to evaluate how well a re-ranking method achieves a trade-off between bias (or any other secondary metric) and the relevancy of individual documents.

We described an evaluation using injection of "polluted" (known biased) documents into a standard news corpus. Our comprehensive evaluation compares various methods for automatically assessing document bias and highlights the effectiveness of BiasRank, particularly when employing BiasScanner, across two information retrieval models with distinct weighting schemes by employing novel and traditional metrics.

In future work, we plan to extend the evaluation setup in order to explore languages other than English. We also would like to collect aggregate bias statistics for entire news outlets in ways similar to Ye and Skiena (Ye and Skiena, 2019), but using our full sentence-level bias analysis (rather than a set of weak proxies like sentiment, as they did); such statistics could then be used as priors to build more comprehensive Web-scale Bayesian models of bias in communication. An integration with fact-checking systems could also be explored. This way, re-ranking could consider not only the bias of a document, as assessable by its linguistic features, but also the factual accuracy of its content.

REFERENCES

Ad Fontes Media (eds.) (2024). Reuters bias and reliability. (accessed 2024-10-15).

Andrzej Białecki, Robert Muir, G. I. (2012). Apache Lucene 4. In *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval Held in Portland, OR, USA, 16th August 2012*, pages 17–24.

Bakshy, E., Messing, S., and Adamic, L. A. (2015). Exposure to ideologically diverse news and opinion on facebook. *Science*, 348(6239):1130–1132.

Bernhardt, L., Dewenter, R., and Thomas, T. (2020). Watchdog or loyal servant? political media bias in us newscasts.

Budak, C., Goel, S., and Rao, J. M. (2016). Fair and balanced? quantifying media bias through crowdsourced content analysis. *Public Opinion Quarterly*, 80(Suppl. 1):250–271.

Costa-Jussà, M. (2019). An analysis of gender bias studies in natural language processing. *Nature Machine Intelligence*, 1:495–496.

Craswell, N., Zoeter, O., Taylor, M., and Ramsey, B. (2008). An experimental comparison of click position-bias models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 87—-94, New York, NY, USA. Association for Computing Machinery.

Da San Martino, G., Yu, S., Barrón-Cedeño, A., Petrov, R., and Nakov, P. (2019). Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*,

pages 5636–5646, Hong Kong, China. Association for Computational Linguistics.

Ekstrand, M. D., Das, A., Burke, R., and Diaz, F. (2022). Fairness in information access systems. *Found. Trends Inf. Retr.*, 16(1-2):1—-177.

Epstein, R. and Robertson, R. E. (2015). The search engine manipulation effect (seme) and its possible impact on the outcomes of elections. *Proceedings of the National Academy of Sciences*, 112(33):E4512–E4521.

Gezici, G., Lipani, A., Saygın, Y., and Yilmaz, E. (2021). Evaluation metrics for measuring bias in search engine results. *Information Retrieval Journal*, 24(2):85–113.

Gharahighehi, A., Vens, C., and Pliakos, K. (2021). Fair multi-stakeholder news recommender system with hypergraph ranking. *Information Processing & Management*, 58(5):102663.

Herman, E. S. and Chomsky, N. (1988). *Manufacturing Consent: The Political Economy of the Mass Media*. Pantheon Books, New York, NY, USA, 1st edition.

Hu, D., Jiang, S., E. Robertson, R., and Wilson, C. (2019). Auditing the partisanship of google search snippets. In *The World Wide Web Conference*, WWW '19, pages 693–704, New York, NY, USA. ACM.

Jaenich, T., McDonald, G., and Ounis, I. (2024). Fairness-aware exposure allocation via adaptive reranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, pages 1504–1513, New York, NY, USA. ACM.

Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada*, pages 133–142. ACM.

Kulshrestha, J., Eslami, M., Messias, J., Zafar, M. B., Ghosh, S., Gummadi, K. P., and Karahalios, K. (2019). Search bias quantification: investigating political bias in social media and web search. *Information Retrieval Journal*, 22(1–2):188–227.

Lauw, H. W., Lim, E.-P., and Wang, K. (2006). Bias and controversy: beyond the statistical deviation. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 2006, pages 625–630, New York, NY, USA. ACM.

Leidner, J. L. and Callison-Burch, C. (2003). Evaluating question answering systems using FAQ answer injection. In *Proceedings of the 6th Annual CLUK Research Colloquium*, CLUK.

Lippmann, W. (1922). *Public Opinion*. Harcourt, Brace & Co., New York. First edition.

Martin, G. J. and Yurukoglu, A. (2017). Bias in cable news: Persuasion and polarization. *The American Economic Review*, 107(9):2565–2599.

Menzner, T. and Leidner, J. L. (2024a). Biasscanner: Automatic detection and classification of news bias to strengthen democracy. Cornell University ArXiv preprint server (accessed 2024-07-30).

Menzner, T. and Leidner, J. L. (2024b). Experiments in news bias detection with pre-trained neural transformers. In *Proceedings of the 46th European Conference in Information Retrieval (ECIR 2024), Glasgow, UK, March 24-28, 2024*, volume IV of *Lecture Notes in Computer Science (LNCS 14611)*, pages 270–284, Cham, Switzerland. Springer Nature.

Menzner, T. and Leidner, J. L. (2024c). Improved models for media bias detection and subcategorization. In *Natural Language Processing and Information Systems:Proceedings of the 29th International Conference on Applications of Natural Language to Information Systems, NLDB 2024 Turin, Italy, June 25–27, 2024, Proceedings, Part I*, volume 14762 of *Lecture Notes in Computer Science, LNCS*, pages 181–196.

Menzner, T. and Leidner, J. L. (2025). Automatic news bias classification for strengthening democracy. In *Proceedings of the 47th European Conference on Information Retrieval (ECIR)*. Accepted for publication.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Ovaisi, Z., Ahsan, R., Zhang, Y., Vasilaky, K., and Zheleva, E. (2020). Correcting for selection bias in learning-to-rank systems. In *Proceedings of The Web Conference 2020*, WWW 2020, pages 1863–1873, New York, NY, USA. ACM.

Paramita, M. L., Kasinidou, M., and Hopfgartner, F. (2022). Base: a bias-aware news search engine for improving user awareness (prototype). In *Biennial Conference on Design of Experimental Search & Information Retrieval Systems*.

Park, S., Kang, S., Chung, S., and Song, J. (2012). A computational framework for media bias mitigation. *ACM Trans. Interact. Intell. Syst.*, 2(2):1–32.

Raj, A. and Ekstrand, M. D. (2022). Measuring fairness in ranked results: An analytical and empirical comparison. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pages 726—-736, New York, NY, USA. ACM.

Ratz, L., Schedl, M., Kopeinik, S., and Rekabsaz, N. (2024). Measuring bias in search results through retrieval list comparison. In *Proceedings of the 46th European Conference on Information Retrieval (ECIR 2024), Glasgow, UK, March 24–28, 2024, Proceedings, Part V*, pages 20–34, Heidelberg, Germany. Springer-Verlag.

Sanfilippo, S. (2009). Redis in-memory data structure server. (accessed 2024-11-04).

Sapiezynski, P., Zeng, W., Robertson, R. E., Mislove, A., and Wilson, C. (2019). Quantifying the impact of user attentionon fair group representation in ranked lists. In *Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 553–562. Association for Computing Machinery (ACM).

Spinde, T., Rudnitckaia, L., Mitrović, J., Hamborg, F., Granitzer, M., Gipp, B., and Donnay, K. (2021).

Automated identification of bias inducing words in news articles using linguistic and context-oriented features. *Information Processing & Management*, 58(3):102505.

Wendelin, M., Engelmann, I., and Neubarth, J. (2017). User rankings and journalistic news selection: comparing news values and topics. *Journalism Studies*, 18(2):135–153.

Wu, C., Wu, F., Qi, T., and Huang, Y. (2022). FairRank: Fairness-aware single-tower ranking framework for news recommendation. Cornell University ArXiv pre-Print Server (accessed 2024-07-08).

Ye, J. and Skiena, S. (2019). Mediarank: Computational ranking of online news sources. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD 2019, pages 2469–2477, New York, NY, USA. ACM.