It's All Relative: Learning Interpretable Models for Scoring Subjective Bias in Documents from Pairwise Comparisons

Aswin Suresh
EPFL, Switzerland
aswin.suresh@epfl.ch

Chi-Hsuan Wu*
HKUST, Hong Kong
cwuau@connect.ust.hk

Matthias Grossglauser
EPFL, Switzerland
matthias.grossglauser@epfl.ch

Abstract

We propose an interpretable model to score the subjective bias present in documents, based only on their textual content. Our model is trained on pairs of revisions of the same Wikipedia article, where one version is more biased than the other. Although prior approaches based on bias *classification* have struggled to obtain a high accuracy for the task, we are able to develop a useful model for *scoring* bias by learning to accurately perform pairwise comparisons. We show that we can interpret the parameters of the trained model to discover the words most indicative of bias. We also apply our model in three different settings by studying the temporal evolution of bias in Wikipedia articles, comparing news sources based on bias, and scoring bias in law amendments. In each case, we demonstrate that the outputs of the model can be explained and validated, even for the two domains that are outside the training-data domain. We also use the model to compare the general level of bias between domains, where we see that legal texts are the least biased and news media are the most biased, with Wikipedia articles in between.

1 Introduction

Subjective bias as defined by Pryzant et al. (2020) is that which "occurs when language that should be neutral and fair is skewed by feeling, opinion, or taste (whether consciously or unconsciously)". With the explosion of human generated data on the web, content affected by such subjective bias inform the perspectives and influence the decisions, both political and otherwise, of an increasing number of people. When people are unaware of bias in present in the content they consume, it contributes to the formation of echo chambers and makes it difficult to build consensus for actions for the common good. Therefore, it is important to identify and measure this bias and to do so in an explainable manner so as to be trustworthy and easy to verify.

Currently, this is done manually in several domains: Wikipedia editors mark articles and edits as violating neutrality, companies such as AllSides (AllSides, 2022) provide ratings of bias in the media, and political scientists analyze speeches to study subjective language as expressions of ideological positions. However, such manual analysis cannot scale to the exponentially growing size of web data, hence necessitating the use of automated approaches. Machine-learning models that can benefit from the large training data are of particular interest in this regard.

The English-language Wikipedia is in many ways an ideal source of training data for these models. It has a neutral point of view (NPOV) policy (Wikipedia, 2022c), the adherence to which can be used as a measure of unbiasedness (neutrality). The policy requires following principles such as not stating opinions as facts (and vice versa), not using language that sympathizes with or disparages the subject, etc. Wikipedia also has an active community of editors that enforces this policy by making edits to reword or remove problematic content from articles and leaving comments to indicate NPOV issues. Moreover, the data is extensive due to Wikipedia's vast collection of articles spanning a wide range of subjects; and the complete revision history of these articles, along with the editors' comments, is accessible to the public.

Our goal in this work is to develop a model trained on POV-related edits to Wikipedia articles that can quantify bias in web documents and study its applicability to Wikipedia itself, as well as to domains outside the training data such as news and legal texts. In addition to being reasonably accurate, we also want the model to be *interpretable*, i.e., we want to use the parameters of the trained model to infer the words indicative of bias and to

*Work done while the author was at EPFL.

Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1341–1353

explain the output of the models.

1.1 Bias Classification versus Scoring

Previous work on bias modeling predominantly considers the task of bias classification, i.e., classifying a given piece of text as biased or unbiased (Pryzant et al., 2020; Zhong et al., 2021; Li et al., 2022). However, we suggest that classifying general web documents in this manner is, for two reasons, not a well-defined task.

First, the threshold for deciding whether a text is biased or not is subjective, especially for longer texts such as documents. In fact, previous work has found poor inter-annotator agreement when obtaining ground-truth labels (Lim et al., 2020; De Kock and Vlachos, 2022; Spinde et al., 2021).

Second, this threshold varies depending on the topic and the domain of the document. For instance, a Wikipedia article considered 'unbiased' on a politically controversial topic is arguably prone to having more subjective statements than a 'biased' one describing an objective scientific truth.

Therefore, in this work, we instead consider the task of assigning a *real-valued score* for the bias in a document. Unlike binary labels such as 'biased' and 'unbiased', a bias score can be assigned without the need for a topic or domain-dependent threshold. Texts from domains/topics prone to greater subjectivity can be assigned a relatively higher bias score *in general*, while also reflecting the level of bias of the specific text relative to other texts in the same domain. For instance, articles in news media could have a higher bias score in general than Wikipedia articles, but a factual news article can still have a lower bias score than an editorial.

Such a real-valued score can be derived from the Bradley-Terry model of pairwise comparisons (Bradley and Terry, 1952) that is trained to predict which text, among a given pair of texts, is more biased. The model uses a score for the items being compared (the texts in this case). This score, when parametrized in terms of domain-independent text representations as features, can be interpreted as a measure of bias that is generalizable across topics and domains.

Note that we are not considering a regression task. We do not use any ground-truth bias scores to train a regression model. Rather, the scores are latent parameters of the bias comparison model which is trained for the pairwise classification task of identifying which text in a pair is more biased. Only the labels for this classification task are observed.

The use of latent scores for words has been explored by Vafa et al. (2020) in the context of scoring political ideology. However, our approach is fundamentally different as it is supervised, using the outcomes of paired comparisons. Similar scores and models have been used, for instance, to quantify the skill of tennis players based on the outcomes of matches between them (McHale and Morton, 2011) or the skill of parliamentarians based on their success in getting their amendments accepted (Kristof et al., 2021).

We can obtain abundant training data for the bias comparison task from the revision history of Wikipedia articles. Each time a Wikipedia editor corrects a POV issue present in an article version, a pair of texts is generated where one text (the version before the correction) is more biased than the other (the version after the correction).

Greater inter-annotator agreement and human accuracy have been found for comparisons than classification when modeling subjective quantities like bias (De Kock and Vlachos, 2022; Aroyo et al., 2019). Pairwise comparisons have also been promoted as a more robust framework for using pretrained LLMs for text ranking tasks (Qin et al., 2023).

To the best of our knowledge, we are the first to develop a model for the task of scoring subjective bias in texts using supervised pairwise comparison data.

1.2 Other Comparisons to Related Work

While previous work has primarily focused on the task of identifying bias in short pieces of text such as words and sentences (Pryzant et al., 2020; Zhong et al., 2021), scoring bias at the document level enables us to benefit from additional context information such as the overall topic of the document.

At the document-level, Wong et al. (2021) predict reliability issues using only metadata features while De Kock and Vlachos (2022) consider the task of promotional tone detection. They use much smaller datasets than ours and achieve relatively low performance for the classification task.

Most prior models are based on deep neural networks (DNNs) hence require significant time and GPU resources for training and inference. In particular for training, the models in Pryzant et al. (2020) and Zhong et al. (2021) need several hours, and the model in De Kock and Vlachos (2022) needs more

than a day. DNNs are also difficult to interpret. Although an explanation can be given for which parts of a *given* text are biased, it is difficult to answer, based on the trained model, which words *in general* are indicative of bias.

Compared to prior bias models, our model is easily interpretable as it avoids using DNNs. It is also relatively inexpensive computationally to train and use while achieving similar or better accuracies. We also study the application of the model in a variety of document domains.

We seek to answer the following research questions:

- **RQ1:** Given a pair of consecutive revisions (versions), of the same Wikipedia article, how well can we *predict* which one among them is more biased, using only their textual content?
- **RQ2:** Can we *understand* which words are correlated with bias?
- **RQ3:** How widely can the bias scores computed by these models *generalize*? Can they measure the evolution of bias across the entire history of an article? Can they compare bias in different articles and in other texts beyond Wikipedia articles?

Towards answering these RQs, we make the following contributions:

- We develop predictive models for bias comparison and compare their accuracy against several baselines.
- We use the parameters of the trained models to compute a bias score for words and use it to discover words that are indicative of bias.
- We use the trained models to compute a bias score for documents and demonstrate its generalizability across time, topics, and domains. As external domains, we focus on news articles and legal texts, as they are generally expected to have higher and lower subjectivity, respectively, than encyclopedia articles.

Finally, we curate new datasets of Wikipedia articles to train and evaluate our models. We release publicly all the datasets and our code[1].

The rest of the paper is structured as follows. In Section 2, we provide details about the datasets we use for this study. In Section 3, we describe the bias model in detail. In Section 4, we evaluate the performance of the model, explore its interpretability, and comment on some potential applications of the model to other domains. We conclude the paper in Section 5.

2 Datasets

We use four datasets in this paper, two of which we collected ourselves. We will now briefly describe the datasets.

2.1 Wikipedia: Article Neutrality

To train and evaluate our model, we curate a new dataset that we call the Wikipedia article neutrality dataset (WAND). The dataset can be viewed as an article-level version of the sentence-level dataset collected in Zhong et al. (2021).

The dataset consists of the text of pairs of revisions of the same Wikipedia article where one revision is more biased than the other. We collect it by going through the revision history of all articles in the English Wikipedia and by collecting a pair of revisions before and after a POV-related edit is made. We identify the POV-related edits by checking for the presence of certain regular expressions in the comments; we use the same list of expressions used in Zhong et al. (2021).

For each revision, we use the `mwparserfromhell` package (Kurtovic, 2022) to parse its *wikitext* as obtained from the MediaWiki API (Wikimedia, 2023). We then apply the text pre-processing steps, followed by Wong et al. (2021) and Pryzant et al. (2020), to keep only the plain text (excluding wikilinks, templates, and tags) from the main content part of the article (excluding the External Links and References sections).

Our final dataset contains 895,957 revision pairs from 358,941 articles.

2.2 Wikipedia: Controversial Issues

As the WAND dataset contains the revisions at only the times of the POV-related edits, we cannot use it to evaluate the performance of our models in measuring bias evolution. Therefore, we construct a new dataset of revisions of the articles mentioned in Wikipedia's *List of Controversial Issues* (Wikipedia, 2022a). The list contains 1,544 articles in total. Wikipedia editors are urged to regularly check these articles to make sure that the

[1]Code and data are at https://github.com/indy-lab/compair

presentation follows the NPOV policy, as they are frequently subjected to biased edits.

For each article, we collect the text for 100 revisions periodically sampled from its history. The text is pre-processed, as in WAND, to retain only the plain text from the main article content.

2.3 News

We use the Webis Bias Flipper-18 dataset (Chen et al., 2018) that contains 6,448 news articles from 77 outlets (mostly from the United States) with different ideological biases (left, right, and center). The articles that describe the same event are grouped into stories, which enables us to eliminate the effect of the event itself by ranking articles within each group. There are 2,781 stories in total.

The grouping of the news articles and the ideological bias labels of the outlets come from AllSides.com (AllSides, 2022). This website aims to present balanced coverage of news by presenting articles from outlets with different ideological biases. The ideological bias labels for each outlet are determined by a combination of factors, including editorial review and community feedback.

2.4 European Parliament: Law Amendments

We use the dataset of amendments proposed in the eighth term of the European Parliament, released by Kristof et al. (2021). Each amendment in the dataset consists of a pair of texts. The first text is a paragraph of the original law text, as drafted by the European Commission. The second text is the amended version of the same paragraph as proposed by a group of parliamentarians when the law is being discussed within the European Parliament.

Each proposed amendment is voted on and it may be (fully or partially) accepted or rejected for incorporation into a modified draft law. The dataset contains 28,407 original texts and 98,245 proposed amendments, out of which 37,689 were fully or partially accepted and 73,604 were fully or partially rejected.

3 Model

We now describe the model we propose for bias comparison and scoring. Interpretability and computational efficiency are our primary concerns, hence we generally avoid using DNNs in the model architecture. Nevertheless, to estimate the performance improvements we can expect from using DNNs we also build a version of our model that uses them at the feature extraction step (see Appendix A).

3.1 Features

To represent the text of a document, we use the normalized sum of the embedding vectors of the words in the text. We use pre-trained fastText (Unsupervised) embeddings (Bojanowski et al., 2017) that were trained on the English Wikipedia.

We obtain the vector representation of a text i as

$$\widehat{\mathbf{t_i}} = \frac{\mathbf{t_i}}{\|\mathbf{t_i}\|}, \quad \widehat{\mathbf{t_i}} \in \mathbb{S}_1^d \text{ (unit sphere)}, \tag{1}$$

where

$$\mathbf{t_i} = \sum_{w \in \mathcal{V}_i} n_i(w) \mathbf{v_w}. \tag{2}$$

Here $\mathcal{V}_i$ is the set of words in text i, $n_i(w)$ is the frequency of word w in text i and $\mathbf{v_w}$ is the embedding vector of the word.

3.2 Model Architecture

Our model takes inputs in the form of *pairs* of texts and predicts which text is more biased than the other. We use the Bradley-Terry model of pairwise comparison outcomes (Bradley and Terry, 1952).

We define the probability that text i is more biased than text j to be

$$P(i \succ j) = \frac{e^{s_i}}{e^{s_i} + e^{s_j}}, \tag{3}$$

where $s_i, s_j \in \mathbb{R}$ are *bias scores* of texts i and j, respectively (higher means more biased).

We model the bias score of a text i as the sum of the bias contributions of the words present in the text, weighted by the number of times each word occurs in the text. More precisely, we define

$$s_i = \frac{1}{\|\mathbf{t_i}\|} \sum_{w \in \mathcal{V}_i} n_i(w) B(w, i), \tag{4}$$

where $B(w, i)$ is the bias contribution of the word w given the topic of text i. We also include a normalizing factor $\|\mathbf{t_i}\|$ to ensure that the bias score of a text does not depend on its length or generality. This enables us to compare the bias within a diverse set of texts. More explanation is provided in Appendix B.

We model the bias contribution $B(w, i)$ as a function of both the word w and the text i, as the bias induced by words can change depending on the topic of the text. For instance, the word *malicious*, when used as an adjective to describe the

nature of a specific person, usually indicates bias, but when used within a computer science article, it can be legitimate (e.g.,*malicious code*).

To model this we define $B(w, i)$ as

$$B(w, i) = \mathbf{f_i}^T \mathbf{v_w}, \tag{5}$$

where $\mathbf{f_i} \in \mathbb{R}^d$ is the bias word *query vector* for text i and $\mathbf{v_w} \in \mathbb{R}^d$ is the embedding vector of word w. The smaller the angle between $\mathbf{f_i}$ and $\mathbf{v_w}$ is, the higher the bias contribution of w given the topic of text i.

The query vector $\mathbf{f_i}$ depends on the topic of text i. We model it as an affine function of the vector representation $\widehat{\mathbf{t_i}}$ of the text i,

$$\mathbf{f_i} = \mathbf{W}^T \widehat{\mathbf{t_i}} + \mathbf{b}, \tag{6}$$

where $\mathbf{W} \in \mathbb{R}^{d \times d}$ and $\mathbf{b} \in \mathbb{R}^d$ are learned parameters. This simple formulation enables us to easily compute a general (topic-independent) version of the word bias score that we describe later.

Substituting (6) in (5), and (5) in (4), and using (1) and (2) to simplify, we get the bias score of the text as

$$s_i = \widehat{\mathbf{t_i}}^T \mathbf{W} \widehat{\mathbf{t_i}} + \mathbf{b}^T \widehat{\mathbf{t_i}}. \tag{7}$$

To interpret the model to identify the bias of words, we need to get the true values of all $B(w, i)$, for which we need precise inference to be possible for $\mathbf{W}$ and $\mathbf{b}$ (i.e., the model should be identifiable). It is straightforward to see that this is satisfied if and only if $\mathbf{W}$ is symmetric. We therefore parameterize $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{U} + \mathbf{U}^T, \tag{8}$$

where $\mathbf{U} \in \mathbb{R}^{d \times d}$ is the variable that is optimized during learning.

While $B(w, i)$ gives the bias contribution of word w when it appears in text i, we are also interested in obtaining the general bias score of a word in a given corpus of texts $\mathcal{C}$ without specifying any particular text. Hence we define the general bias score of a word w as an average of its bias score over all texts, i.e.,

$$GB(w) = \frac{\sum_{i \in \mathcal{C}} B(w, i)}{|\mathcal{C}|} = \bar{\mathbf{t}}^T \mathbf{W} \mathbf{v_w} + \mathbf{b}^T \mathbf{v_w}, \tag{9}$$

where

$$\bar{\mathbf{t}} = \frac{\sum_{i \in \mathcal{C}} \mathbf{t_i}}{|\mathcal{C}|}. \tag{10}$$

Note that the affine formulation of $\mathbf{f_i}$ enables us to compute $GB(w)$ by averaging the text representations $\mathbf{t_i}$ separately, thereby reducing the computational complexity.

We call a version of our model including only the linear term $\mathbf{b}$ in (6) as *ComPair-Linear* model and the full model including both terms as the *ComPair-Quadratic* model.

3.3 Training

We use the WAND dataset for training. We split the revision pairs into training, validation, and test sets in the ratio 90:5:5. To avoid data leakage, we take care to ensure that all pairs from a given article are present in the same split.

We train each model by maximizing the likelihood of the training data, under the probability model in (3). More precisely, we solve the optimization problem given by

$$\max_{\theta} \prod_{(i,j) \in \mathcal{D}} P(i \succ j | \theta), \tag{11}$$

where $\theta = \{\mathbf{U}, \mathbf{b}\}$ is the set of parameters to be learned, $(i, j) \in \mathcal{D}$ are the revision pairs in the train set (i is the version before the edit, j is the version after the edit), and $P(i \succ j | \theta)$ is the probability that i is more biased than j given the parameters θ, modelled as in (3).

We use mini-batch stochastic gradient ascent for the maximization. Models take approximately 2 hours to train. We do not observe any overfitting based on the performance of the model on the validation set and therefore do not use any regularization.

4 Evaluation and Applications

In this section, we evaluate the performance of our models, examine their interpretability, and explore their applications in a variety of domains. Some additional analysis is also given in the appendix.

4.1 Evaluation

We evaluate the ability of our models to perform pairwise comparisons of bias by measuring their accuracy on the test set.

We compare against several baselines which we describe below:

- **Random**: The random classifier predicts one of the two versions in a pair uniformly at random to be the more biased one.

Model	Accuracy(%)
Random	50 ± 0.46
Words2Watch	63.4 ± 0.44
DeepSentClass-Linear	68.35 ± 0.43
DeepSentClass	76.01 ± 0.39
ComPair-Linear	75.29 ± 0.40
ComPair-Quadratic	$\mathbf{76.84 \pm 0.39}$
Human	74.00 ± 8.60

Table 1: Accuracy of models

- **Wiki Words to Watch (Words2Watch)**: Wikipedia maintains a list of words that could potentially cause bias called *Words to Watch* (Wikipedia, 2022b). This model compares the versions using the count of such words in the text.

- **Sentence Classifier (DeepSentClass)** These models are based on the sentence-level bias classification models using DNNs developed by Zhong et al. (2021). They use a BERT model (Devlin et al., 2019) finetuned on sentence pairs before and after an NPOV edit, to classify whether a given sentence is biased or not. DeepSentClass compares the versions using the mean of the predicted bias probability for the sentences in each version[2]. To have a more fair comparison with our method in terms of training cost, we also use a version of DeepSentClass, which we call DeepSentClass-Linear, where the DNN weights are kept fixed and only the linear layer is trained.

The test accuracy of all baseline models and our models, and their 95% confidence intervals are given in Table 1. We also include a human performance benchmark which was obtained by one of the authors manually labeling 100 randomly chosen pairs from the test set.

ComPair-Quadratic achieves 76.84% accuracy, significantly outperforming the baselines. Remarkably, it performs similarly to *DeepSentClass* which requires significantly more resources for training and inference[3].

The higher accuracy achieved by *ComPair-Quadratic* relative to *ComPair-Linear* suggests that the information given by the document topic in computing $B(w, i)$ is beneficial. We use *ComPair-Quadratic* in all our subsequent experiments.

[2]We tried using the maximum as well, but it had significantly worse performance.

[3]Averaged over 1,000 random pairs from the test set, DeepSentClass needs 1,278ms for inference on a GPU while ComPair-Quadratic needs 130ms on a CPU.

High $GB(w)$		**Low** $GB(w)$
1-10	**11-20**	$\mathbf{P_{10\%}}$
impressive	stunning	waived
finest	horrible	readings
superb	splendid	discussed
wonderful	talented	convened
toughest	amazing	attended
formidable	pleasing	supplements
brilliant	proud	chaired
exciting	fascinating	grams
beautiful	clever	served
excellent	terrible	suggested

Table 2: Words w in decreasing order of $GB(w)$

The model we described and evaluated so far is our primary model that uses fastText (Unsupervised) word embeddings. To estimate the performance improvement we can expect from using DNNs, we also build and evaluate a version of *ComPair-Quadratic* that uses contextual word embeddings (where the embedding is different for each occurrence of a word), that we call *ComPair-Quadratic-DNN*. For the pairwise bias comparison task, *ComPair-Quadratic-DNN* achieves an accuracy of 77.56 ± 0.38 on the test set which is comparable with *ComPair-Quadratic*. However, the inference time is significantly higher. More details may be found in Appendix A.

4.2 Interpretation

We interpret the parameters of the trained model to see the words indicative of bias.

First, we obtain the general bias score $GB(w)$ for every word w in the WAND dataset. The list of top 20 words with the highest $GB(w)$, and the list of 10 words at the 10^{th} percentile are given in Table 2.

We see that the words with the highest scores are typically subjective adjectives and other subjective words. The words with lower scores are typically verbs and common nouns.

We can also compare the values of $B(w, i)$ for the same word in different articles to see how the bias induced by the word changes depending on the article's topic. For instance, the word *poorly* when used in the sense of bad performance in sports (in the article *Howard Johnson* (baseball player)) has a

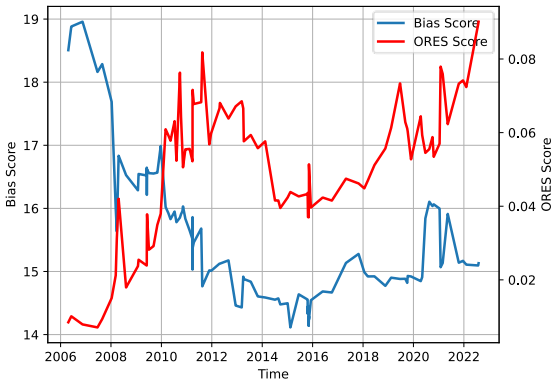

Figure 1: Bias score of *Heritability of IQ* over time. Wikipedia ORES article quality scores are plotted for comparison. Spearman correlation: -0.27, p-value 0.008.

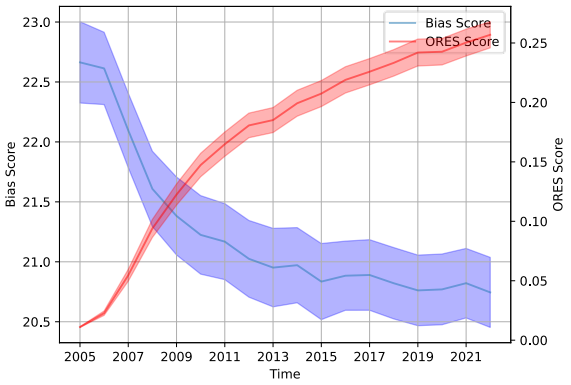

Figure 2: Average bias score and ORES score of articles over time. The dark lines are the scores and the shaded area indicates the 95% confidence interval.

$B(w, i)$ score of 324.12. In contrast, it has a much lower bias score of 30.56 when used to describe something 'burning poorly' in the article *Hydrogen Storage*.

We now comment on some applications of our model for scoring bias in different settings. Note that the model has only been trained on Wikipedia data.

4.3 Temporal Evolution of Bias

We first apply the model to study the temporal evolution of bias by plotting the bias score s_i over time as revisions are made to an article. We use the *Wikipedia: Controversial Issues* dataset for the analysis in this section.

As a typical example, we show the plot of the article *Heritability of IQ* in Figure 1.

For comparison, we also plot in the same figure the article quality score computed by the Wikipedia Objective Revision Evaluation Service (ORES) (Halfaker and Geiger, 2020)[4].

We see from Figure 1 that the bias score computed by our model has a negative correlation with the ORES score, which is expected as bias negatively affects quality. The median Spearman correlation across all articles in the dataset is -0.27.

In addition to the evolution of bias for individual articles, we also study the trend of the average bias across articles over time. This would help to answer questions such as whether on average the bias of an article decreases over time in Wikipedia and if so how fast it decreases.

We consider all articles in the dataset that were created around the same time (in 2003 or 2004), and average each of their bias scores at the same points in time throughout their history. We get the trend shown in Figure 2.

We can clearly see that on average the bias of an article decreases over time until it reaches a steady state and that it reaches this state in about ten years. The increasing trend of the ORES score also supports this conclusion.

4.4 Media Bias

We now apply the model to score bias in the domain of news media, a different domain from its training domain of Wikipedia. We use the News Dataset in this analysis.

We estimate the relative bias level of different outlets to rank them and identify the ones that are most and least biased. First, we obtain a bias score for the articles from each outlet using our trained model. For every news story, we order the articles covering the story in terms of the bias score and compute the percentile bias score for each article in the story. Finally, we compute the average of the percentile bias scores of the articles from a news outlet to get the mean percentile bias score of the outlet.

We plot the mean percentile bias scores of the outlets along with their 95% confidence intervals in Figure 3. For clarity, we only show in the plot the 6 outlets with the smallest confidence interval from each category (left, right, and center) and the mean scores.

[4]ORES uses a machine learning model to predict article quality, based primarily on structural features. Editors use these assessments to identify the articles to focus on. The ORES score for a revision can be obtained by querying a public API (Wikipedia, 2023).

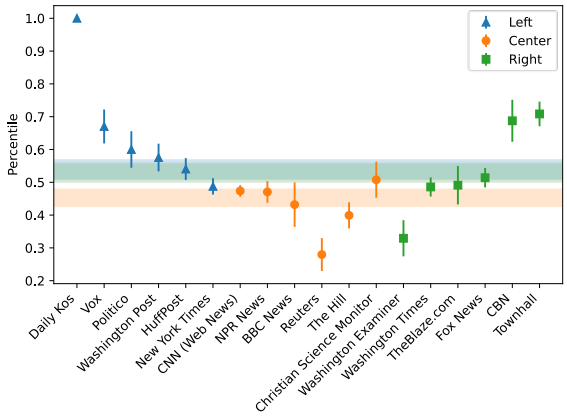

Figure 3: Mean percentile bias score of news outlets. The bands show the confidence interval of the mean percentile bias score for left, right and center articles. These include articles from outlets not shown in the figure.

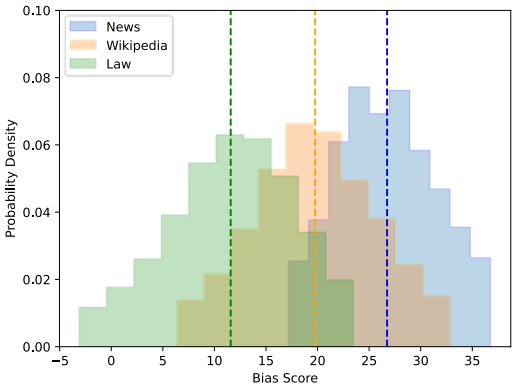

Figure 4: Distribution of bias scores across domains

Legal text	**Mean Score**
All (Original + Amendments)	11.34 ± 0.04
Original	9.70 ± 0.10
Amendments	11.81 ± 0.05
Amendments (Accepted)	11.64 ± 0.08
Amendments (Rejected)	12.01 ± 0.06

Table 3: Mean bias score of legal texts.

Although there is overlap between individual outlet scores, we see from the confidence intervals of the mean scores that articles from center outlets have significantly lower mean score than those from left and right outlets. Looking at the individual outlet scores, we see that the *Reuters* news agency which is known for its policy of objective language has the lowest bias score. *The Hill*, which claims to provide "objective" and "non-partisan coverage", also has a relatively low bias score. On the other hand, outlets like *Daily Kos* on the liberal side and *Townhall* on the conservative side are open about their political bias. Their articles commonly include partisan commentary on news events and consequently have a very high bias score. *Washington Examiner* is an outlier; it is considered by AllSides to have a Lean-Right bias but has a quite low mean bias score. On manually examining their articles in our dataset, we find that bias occurs in the form of giving a greater fraction of coverage to certain views, rather than word choice or other forms of subjective language. Our model is not expected to detect such forms of bias which explains the low bias score.

Finally, we plot the distribution of bias scores of all the news articles in Figure 4, along with the distribution of scores in Wikipedia. We see that the scores are generally higher, as news articles frequently contain subjective commentary on events, while this is disallowed in Wikipedia.

4.5 Bias in Legal Texts

We use the European Parliament Amendments dataset to study bias scoring in the legal domain. We give in Table 3 the mean bias scores of the different subsets of legal texts in the dataset.

First, we see that the magnitude of bias scores is significantly lower than that of Wikipedia, as is also clear from the distribution of bias scores in Figure 4. This is the opposite of what was observed in the case of News. This could be due to the fact that legal provisions are carefully crafted to be objective so as to minimize ambiguity in the interpretation of the law, while they also tend to avoid partisan language in the introduction sections so that the text is palatable to legislators of diverse political leanings.

Interestingly, we see that the average bias of the amendments that the parliamentarians propose is higher than that of the original text proposed by the commission. On manually examining the amendments with the highest difference in bias scores, we see that many of them change the introductory sections of the law (explanatory memoranda, recitals etc.) by introducing partisan and subjective language. Nevertheless, we see that among the proposed amendments, the ones that get accepted have relatively a smaller bias on average.

5 Conclusion

In this paper, we developed a simple, interpretable model to score bias in documents by learning from pairwise comparisons. We curated two novel datasets based on Wikipedia revision histories to train and evaluate our model. Formulating the problem as assigning a real-valued score for bias, rather classifying a text as biased or unbiased, reduces subjectivity and issues of thresholding. We obtain strong performance on a holdout set of pairs of Wikipedia revisions.

Importantly, the model is interpretable: we can score individual words, a feature that an editor might rely upon to quickly identify the most problematic parts of a document that contribute to the bias. The list of globally most biased words contains a convincing list of strong adjectives and terms that tend to express emotions.

We explored the predictions of the model over datasets including news articles and law amendments. The bias distributions over the three domains (Wikipedia, news, laws) are quite different, with news the most biased, and laws the least, which can be explained by the policies governing the creation of content in each of them. We also observe that we can score the bias in different news outlets; these scores align well with crowdsourced labelings of bias of these outlets.

The model we developed can be integrated into applications to identify, measure, and monitor bias. For instance, one could build a browser extension to enable users to identify bias in online documents and thereby guard themselves against undue influence. Authors of documents that are expected to use objective language (such as legal documents or scientific articles) can measure the bias score to guide their writing. Wikipedia and news editors could monitor bias as revisions are made to articles so as to take corrective action when needed.

Ultimately, we expect this work to contribute to better identifying and correcting both deliberate and subconscious bias in online discourse.

6 Broader Impact, Limitations, and Ethical Considerations

In addition to the bias scoring model we developed, which is applicable in a wide variety of domains, the methodology that we adopted of casting bias as a relative quantity and learning from pairwise comparisons can be extended to a much broader set of problems in natural language processing. It is particularly suited to those settings where the threshold for absolute categorization may be subjective or depends on many factors, while there is more agreement in comparisons. Examples include measuring hateful content, agreeableness, humor, sentiment, etc. While learning from pairwise comparisons is being increasingly applied recently to many NLP tasks, we would only like to draw attention to the fact that there are several tasks for which this is still not applied to the best of our knowledge (hate speech being one example) and we hope that our work can join other similar efforts in inspiring further future research in this direction.

All data we use in this work is from publicly available sources. Wikipedia data that we collect is publicly released under the CC BY-SA and GFDL licenses and analysis of this content does not require informed consent.

Machine learning models are limited by the data that they learn from. Therefore our models inherit any bias that is inherent in Wikipedia's neutrality policy or the manner in which the editors interpret and enforce that policy. An editorial decision that is made based on the output of these models could also serve to reinforce such bias. However, the interpretability of our models mitigates this risk to some extent. For instance, if the model generates an unexpected output an editor can obtain the words that contributed to the model's assignment of a high or low bias score and perform an informed reassessment.

Our models are designed to measure subjectivity in language, but there are several other kinds of bias such as selection bias (giving a greater fraction of coverage to certain views) or demographic bias that are not within its scope. The models also cannot distinguish between truth and hoax, hence it will assign a low bias score to a false statement that uses objective language.

Acknowledgments

We thank Weier Liu who contributed during the initial phase of this project. We also thank Holly Cogliati-Bauereis and the anonymous reviewers for careful proof-reading and constructive feedback.

References

Allen Institute for AI. 2022. longformer-base-4096. https://huggingface.co/allenai/longformer-base-4096. Accessed: 2022-08-1.

AllSides. 2022. AllSides.com. https://www.allsides.com/unbiased-balanced-news. Accessed: 2022-09-15.

Lora Aroyo, Lucas Dixon, Nithum Thain, Olivia Redfield, and Rachel Rosen. 2019. Crowdsourcing subjective tasks: The case study of understanding toxicity in online discussions. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 1100–1105, New York, NY, USA. Association for Computing Machinery.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document Transformer. *arXiv preprint arXiv:2004.05150*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Wiki word vectors. https://fasttext.cc/docs/en/pretrained-vectors.html.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Wei-Fan Chen, Henning Wachsmuth, Khalid Al-Khatib, and Benno Stein. 2018. Learning to flip the bias of news headlines. In *11th International Natural Language Generation Conference (INLG 2018)*, pages 79–88, Tilburg, Netherlands. Association for Computational Linguistics.

Christine De Kock and Andreas Vlachos. 2022. Leveraging Wikipedia article evolution for promotional tone detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5601–5613, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional Transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Aaron Halfaker and R Stuart Geiger. 2020. ORES: Lowering barriers with participatory machine learning in Wikipedia. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–37.

Victor Kristof, Aswin Suresh, Matthias Grossglauser, and Patrick Thiran. 2021. War of words II: Enriched models of law-making processes. In *Proceedings of the Web Conference 2021*, WWW '21, page 2014–2024, New York, NY, USA. Association for Computing Machinery.

Ben Kurtovic. 2022. mwparserfromhell. https://github.com/earwig/mwparserfromhell. Accessed: 2022-10-14.

Zhuoyan Li, Zhuoran Lu, and Ming Yin. 2022. Towards better detection of biased language with scarce, noisy, and biased annotations. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 411–423.

Sora Lim, Adam Jatowt, Michael Färber, and Masatoshi Yoshikawa. 2020. Annotating and analyzing biased sentences in news articles using crowdsourcing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1478–1484, Marseille, France. European Language Resources Association.

Ian McHale and Alex Morton. 2011. A bradley-terry type model for forecasting tennis match results. *International Journal of Forecasting*, 27(2):619–630.

Reid Pryzant, Richard Diehl Martinez, Nathan Dass, Sadao Kurohashi, Dan Jurafsky, and Diyi Yang. 2020. Automatically neutralizing subjective bias in text. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):480–489.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, et al. 2023. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*.

Timo Spinde, Lada Rudnitckaia, Jelena Mitrović, Felix Hamborg, Michael Granitzer, Bela Gipp, and Karsten Donnay. 2021. Automated identification of bias inducing words in news articles using linguistic and context-oriented features. *Information Processing & Management*, 58(3):102505.

Keyon Vafa, Suresh Naidu, and David Blei. 2020. Text-Based Ideal Points. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5345–5357, Online. Association for Computational Linguistics.

Wikimedia. 2023. MediaWiki Action API. https://www.mediawiki.org/wiki/API:Main_page. Accessed: 2023-07-30.

Wikipedia. 2022a. Wikipedia: List of controversial issues. https://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues. Accessed: 2022-08-01.

Wikipedia. 2022b. Wikipedia:Manual of StyleWords to Watch. https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Words_to_watch. Accessed: 2022-08-01.

Wikipedia. 2022c. Wikipedia:NPOV. https://en.wikipedia.org/wiki/Wikipedia:Neutral_point_of_view. Accessed: 2022-10-14.

Wikipedia. 2023. ORES. https://www.mediawiki.org/wiki/ORES. Accessed: 2023-05-15.

KayYen Wong, Miriam Redi, and Diego Saez-Trumper. 2021. Wiki-Reliability: A large scale dataset for content reliability on Wikipedia. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2437–2442, New York, NY, USA. Association for Computing Machinery.

Yang Zhong, Jingfeng Yang, Wei Xu, and Diyi Yang. 2021. WIKIBIAS: Detecting multi-span subjective biases in language. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1799–1814, Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Contextual Word Embeddings

In Section 3 we described our primary models that use fastText (Unsupervised) word embeddings. These embeddings are *static* in the sense that a word has a fixed embedding irrespective of where it occurrs in the text. Contextual word embeddings, on the contrary, represent the meaning of a word in the context where it appears, hence each *occurrence* of the word (a *token*) is represented by a single vector. The embeddings are generated dynamically from the context by a pre-trained DNN. To estimate the performance improvement we can expect from using DNNs, we also build and evaluate a version of *ComPair-Quadratic* that uses contextual word embeddings, that we call *ComPair-Quadratic-DNN*. The text representations $\mathbf{t_i}$ is obtained in a similar manner as for *ComPair-Quadratic* except that we consider tokens instead of words. While training the model, we keep the DNN weights fixed.

The BERT model (Devlin et al., 2019) is arguably one of the most commonly used contextual embeddings and has been used in prior work in bias modeling at the sentence level (Zhong et al., 2021). However, it can model sequences only up to a maximum length of 512 tokens due to the quadratic complexity of the attention mechanism, hence cannot effectively model long documents such as Wikipedia articles.

Therefore, we use a pre-trained Longformer model (Beltagy et al., 2020), which is a variation of BERT that uses sliding window attention, thus enabling it to model long sequences efficiently. Specifically, we use the `longformer-base-4096` model from HuggingFace (Allen Institute for AI, 2022). It has been used to model Wikipedia articles in prior work (De Kock and Vlachos, 2022).

For the pairwise bias comparison task, *ComPair-Quadratic-DNN* achieves an accuracy of 77.56 ± 0.38 on the test set which is comparable with *ComPair-Quadratic*. However, the inference time for 1,000 random pairs is significantly higher (816ms vs. 130ms).

B Normalization Factor

Let K_i be the normalization factor in Equation 4. We then have

$$s_i = \frac{1}{\|\mathbf{t_i}\|} \sum_{w \in \mathcal{V}_i} n_i(w) B(w, i), \tag{12}$$

Substituting (6) in (5), and (5) in (12), and using (1) and (2) to simplify, we get the bias score of the text as

$$s_i = \frac{\|\mathbf{t_i}\|(\widehat{\mathbf{t_i}}^T \mathbf{W} \widehat{\mathbf{t_i}} + \mathbf{b}^T \widehat{\mathbf{t_i}})}{K_i}. \tag{13}$$

We can see from (2) that the quantity $\|\mathbf{t_i}\|$ depends on the total number of words in the text. If a text is concatenated with itself, $\|\mathbf{t_i}\|$ will increase even though the content and bias of the text do not change.

Also, if two texts i and j are similar (i.e., $\widehat{\mathbf{t_i}}$ and $\widehat{\mathbf{t_j}}$ have high similarity) and therefore should have similar bias, but i is more specific and uses a less diverse set of words than j (i.e., the embeddings $\mathbf{v_w}, \forall w \in \mathcal{V}_i$ have a lower variance than the embeddings $\mathbf{v_x}, \forall x \in \mathcal{V}_j$), then $\|\mathbf{t_i}\|$ tends to be larger than $\|\mathbf{t_j}\|$. This could happen for instance if j gives some context around the topic, placing it within a more general topic.

Since we would like the bias score of the text to not change in these cases, we define the scaling factor to be $K_i = \|\mathbf{t_i}\|$. We then have

$$s_i = \widehat{\mathbf{t_i}}^T \mathbf{W} \widehat{\mathbf{t_i}} + \mathbf{b}^T \widehat{\mathbf{t_i}}. \tag{14}$$

Word Type	Mean $GB(w)$
All	48.84 ± 0.28
Words2Watch	108.21 ± 14.30

Table 4: Mean $GB(w)$ of all words vs Words2Watch

C Additional Analyses: Interpretation

To have a more comprehensive analysis of the general word bias scores, we plot in Figure 5 the part-of-speech (POS) distribution of the top 1,000 words in terms of $GB(w)$ in comparison to that of all words. We see clearly that the proportions of adjectives (ADJ) and adverbs (ADV) in the bias-inducing words are significantly higher than that of all words, while the proportion of proper nouns (PROPN) and common nouns (NOUN) are significantly lower. The proportion of verbs (VERB) is nearly the same.

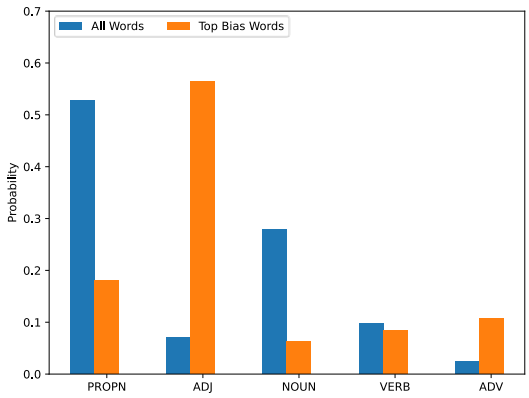

Figure 5: Comparison of POS distributions

To provide external validation for the word bias scores $GB(w)$ generated by the model we rely on the Wikipedia *Words to Watch* list. In Table 4, we give the mean $GB(w)$ of all words as well as the words in the *Words to Watch* list, with their 95% confidence intervals. We see clearly that mean $GB(w)$ of *Words to Watch* is significantly higher than that of all words.

The *ComPair-Quadratic-DNN* model can also be interpreted to identify words and especially multi-word phrases that induce bias. An example is shown in Table 5, where the model correctly identifies the bias-inducing phrase *without a doubt*, which is also mentioned as part of Wikipedia's *Words to Watch*. The *ComPair-Quadratic* model fails to identify the phrase and incorrectly identifies *amp* to be a bias word.

Another change was that apart from no drummer appearing on the album all guitars were recorded directly into the mixing desk without a guitar amp. This is **without a doubt the most** brutal album **ever** made without a drumkit and guitar amp. The spontaneity brought the focus away from feats of musicianship and sent it towards monstrous sounding riffs and **great** songs.

Table 5: An excerpt from the article *The Berzerker*, a death metal band. Words with the highest bias according to the *ComPair-Quadratic-DNN* model are highlighted in bold. The highest bias words according to the *CompPair-Quadratic* model are underlined.

Highest mean s_i	Lowest mean s_i
Anti-Italianism	Macedonia
Patriotism	National Rifle Association
Anti-Irish racism	CBC News
Genocide denial	Federal Marriage Amendment
Black Supremacy	Russian Interference...

Table 6: Most and least biased articles in the *Politics and Economics* section

D Additional Analyses: Wikipedia Article and Topic Bias

In this section, we apply the model to compute bias scores for articles and topics in the Wikipedia: Controversial Issues dataset.

D.1 Article-level bias

First, we compute the average bias score of each article across its revisions and identify the articles with the highest and lowest scores. The results for the articles within the *Politics and Economics* section of the dataset are given in Table 6. We see that the articles with the highest scores are about subjective topics like different '-ism's, and highly controversial topics like racism and denial of genocide. By comparison, the articles with the lowest scores tend to be about fairly objective topics (although still controversial, as we are comparing within the list of controversial topics) like Macedonia, CBC News, and the National Rifle Association. The article on Russian interference in US elections, although it deals with a controversial topic, is well-sourced and protected.

D.2 Topic-level bias

Second, we compare the distributions of bias scores of articles in two different topics, namely *Science, biology, and health*, a relatively objective topic, and *Sex, sexuality, and gender identity* which contains articles on highly controversial topics such as gay rights. The distributions are given in Figure 6. The vertical bars show the positions of the means.

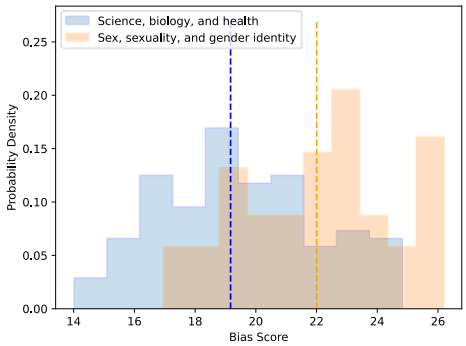

Figure 6: Distribution of bias scores within topics.

We see that the articles in the *Gender identity* topic generally have a higher bias score. There is some overlap as many articles such as *Abortion*, *AIDS*, etc. occur in both topics.