Political Bias Prediction Models Focus on Source Cues, Not Semantics

Selin Chun[1], **Daejin Choi**[2 * †], **Taekyoung Kwon**[1 *]

[1]Department of Computer Science and Engineering, Seoul National University, Republic of Korea
[2]Department of Computer Science and Engineering, Incheon National University, Republic of Korea
slchun@mmlab.snu.ac.kr, djchoi@ewha.ac.kr, tkkwon@snu.ac.kr

Abstract

Significant efforts have been made to analyze the political stance or bias in news articles, especially as political polarization intensifies over the years. Recent advancements in machine learning have enabled researchers to develop various bias prediction models, which typically learn features not only from the text of the news articles but also from external knowledge. However, when training these models, the political bias label assigned to a news article is often based solely on the news source which published it. This approach can be problematic, as a news outlet with a particular political stance might publish an article that reflects a different political perspective.

To address this issue, we first identify distinct text patterns associated with specific news sources or publishers, that are minimally relevant to predicting the political bias of a news article. We then conduct comprehensive experiments to investigate (i) whether existing models trained to predict political bias can also accurately predict the source, and (ii) whether these models change their predictions when a distinct pattern from a source with a different political stance is incorporated into a news article. Our experimental results reveal that all existing models tend to predict the source, even when trained solely to predict bias. Based on these findings, we propose a new deep learning model for political bias prediction that avoids learning source-indicative patterns specific to a given news source.

1 Introduction

With the recent rise in polarized news articles (Auburn 2023) and the increasing polarization of traditional media (Roscini 2021), understanding the perspectives behind journalistic texts has become crucial to avoiding one-sided news. Historically, people relied on a few media channels curated by reporters and editors. Today, however, many turn to digital platforms such as social media, search engines, and news aggregators (Barthel et al. 2020). While the impact of these online platforms on news consumption partisanship is debated (Kitchens, Johnson, and Gray 2020; Dubois and Blank 2018; Calice et al. 2023), classifying the political positions of news articles can serve as a valuable tool to mitigate echo chambers (Cinelli et al. 2021) and filter bubbles (Pariser 2011).

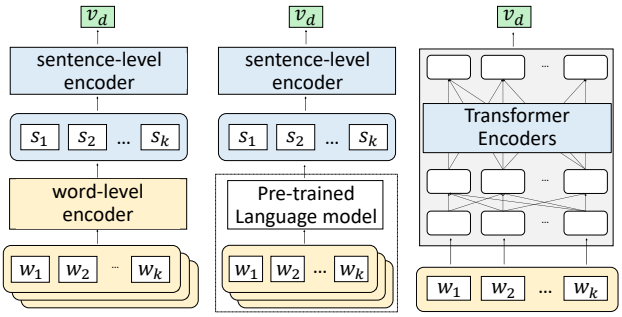
Recently, efforts have been made to address the bias prediction problem in news articles as a document classification task, focusing on analyzing the bias based on the article's text. In Li and Goldwasser (2019), the authors proposed using a Hierarchical LSTM (HLSTM) model to embed the document, combined with a Graph Neural Network (GNN) architecture that considers both document embeddings and article sharing activities to predict political bias. In 2021, they enhanced their model by incorporating entity mentions and political frame indicators (Li and Goldwasser 2021). Following this, other studies have aimed to improve bias classification performance by embedding additional features. For example, Zhang et al. (2022) introduced a GNN architecture that leverages paragraph embeddings from pre-trained RoBERTa and proposed a method to integrate external knowledge into the news text representations. On the other hand, Ko et al. (2023) proposed an end-to-end hierarchical attention-based model that does not rely on a pre-trained language model. They also introduced a method to build and integrate embeddings from multiple knowledge graphs, some of which are specifically designed to reflect political knowledge. By combining these features, prior models have achieved over 90% accuracy in classifying the biases of news articles into one of three categories (left, center, or right), with the text-only model alone reaching nearly 85% accuracy.

Despite achieving high performance in document-level news bias classification, there is a misalignment between the task's objective and the available dataset. Specifically, the bias classification task aims to predict the bias of individual news articles, but these articles are labeled based on the political bias of their source (i.e., the news outlet). As a result, when an article is analyzed, the model tends to identify its source rather than analyze its semantic content. More precisely, when trained on news articles with source-based bias labels, the model tends to learn 'cues' that indicate their sources rather than focusing on the more challenging task of understanding the contextual semantics related to bias.

To address such challenges, we first conduct a thorough analysis of the cues present in the dataset by applying an attribution method to the trained model. This analysis reveals

*Corresponding Authors
†The author has permanently moved to Department of Artificial Intelligence, Ewha Womans University, Seoul, Korea.

distinct patterns specific to individual news sources; for example, some articles from `Politico` include the sentence *"You can unsubscribe at any time."*. These cues vary in form, ranging from full sentences (e.g., subscription solicitations) to phrases, words, and even punctuation marks. We then conduct experiments to determine whether textual deep learning models predict political bias by first identifying the source of a given article. Our analysis shows that when text classification models are trained using source-based bias labels, they often learn to identify the source itself through superficial cues rather than detecting actual political bias through semantic analysis. To address this limitation, we propose a novel model architecture specifically designed to ignore source-specific features, resulting in more reliable political bias detection.



(a) Full-hierarchical (b) Sentence-level (c) Pre-trained

Figure 1: Baseline model architectures categorized as three types

2 Preliminaries

Political Bias Prediction: Problem Formulation

In recent studies, predicting political bias has been treated as a document classification problem. That is, given a news article, the model computes an embedding of text features that may include important signals for the prediction task and assigns the article a bias class (e.g., left, center, or right) based on the embedding. A few studies further adopted additional information such as some external knowledge or social activities such as article sharing in social media platforms.

Prior Datasets for Training/Evaluation

Most prior work rely on either of two datasets: Li and Goldwasser (2019) or Baly et al. (2020), which consists of a large number of news articles, published in multiple news sources, with their associated bias labels. The articles in both datasets were collected from news aggregation websites and a (political bias) label of a given article was assigned as a pre-defined bias label of the news media that published the article. Note that the pre-defined political bias of all the news media are provided by `AllSides.com`, who rates the degrees of political bias of the *media sources* in a 5-point scale, which are left, lean left, center, lean right, and right.

According to `AllSides.com`, the bias rating of a news channel is determined through a combination of methods, including blind bias surveys, editorial reviews, and community feedback. Although the website provides bias labels on a 5-point scale, (Li and Goldwasser 2019) and (Baly et al. 2020) simplified this by aggregating the biases into three categories: left, center, and right.

Baseline Models for Political Bias Prediction

Based on the two datasets, several studies have suggested different text models regarding how to embed a news article. Since news articles are often long documents containing a number of sentences, quadratic requirements of the token-level attention mechanism makes it difficult to process the whole document. For instance, news articles usually does not fit into the 512-token context window of pre-trained language models such as BERT.

Thus, most of prior studies employ hierarchically structured models (Li and Goldwasser 2019, 2021; Feng et al. 2022; Zhang et al. 2022; Ko et al. 2023). That is, as shown in Figures 1a and 1b, each sentence is encoded into a single embedding at the lower level, and a news article consists of the embeddings (from its sentences) at the higher level. At the lower level, the word-level encoder processes words or tokens in a sentence and aggregates them (e.g., by pooling) to create a sentence embedding. For instance, (Li and Goldwasser 2019) uses the LSTM encoder that sequentially encodes words, and then takes the average of all hidden states of each word to create the sentence embedding, while (Ko et al. 2023) takes the similar approach but uses the transformer encoder (TE) instead of the LSTM encoder.

Note that there have been other attempts that try to take advantage of pre-trained language models. In (Feng et al. 2022) and (Zhang et al. 2022), the authors used pre-trained RoBERTa models to encode each sentence into an embedding. After creating the embedding for each sentence, at the higher level, the sentence-level encoder is again used to process the sentence embeddings and to aggregate the result to create the embedding of the document.

3 Identifying Source Cues

Let us analyze whether and how the models in the prior work behave depending on the source cues, which are defined as the distinct text patterns of individual news sources (i.e., some patterns tend to appear only in a particular news source). In this section, we first describe the methodology for the above analysis, and then discuss its results.

Target Dataset

Among the public datasets for political bias prediction mentioned in Section 2, we choose the dataset provided by Baly et al. (2020), which consists of news articles and their corresponding news media. Note that another popular dataset provided by Li and Goldwasser (2019) contains only the metadata of news articles, and the texts of the articles are not available. After inspecting the dataset Baly et al. (2020), we excluded articles that were too long, too short, or duplicates. After the filtering process, we finally obtain 27,327 articles, whose statistical details are summarized in Table 1.

	Left	Center	Right
# of articles	9,315	7,207	10,805
# of sources ($\geq$ 100 articles)	10	6	8

Table 1: Dataset statistics

Identification Method

To find source cues from the dataset, we design a pipeline consisting of two phases: (i) extracting source-indicative sentences from individual news articles and (ii) finding the group of sentences that share similar patterns across all the articles of each source.

In the first phase, we try to retrieve the sentences that may include indicative signals (or patterns) for a specific news source. Thus, we have a *source identification task*, which estimates a source (i.e., a news media) for a given news article. For this, we design a simple deep learning model for this task, which consists of the embedding layers followed by a simple linear layer for a final decision. (Note that, here, we only train the model for sources with 100 articles.) For the embedding layers of the model, we rely on the Hierarchical Attention Network (HAN), which can capture and compute both the word-level and sentence-level features from a given text with a hierarchical architecture. Note that HAN was also employed in (Ko et al. 2023) for political bias prediction. The trained model showed 84% accuracy, demonstrating that the suggested model based on the text features of the news articles performs well.

We next extract source-indicative sentences by identifying distinct sentences which the suggested model mainly focuses on. To this end, we attribute individual sentences in a news article by calculating the gradient$\times$the embedding vectors of individual sentences. That is, after training the suggested model, we first compute the gradient of a given input (i.e., news article) through a backpropagation process until the reverse flow reaches a sentence embedding in HAN. We then multiply the gradient by the embedding vector of each sentence. We sum the elements in the multiplied vector to a single value, which represents a saliency of a sentence in an article. Based on the attribution scores, we select the sentence with the highest score. Note that this extraction process is iterative. We remove the selected sentence from the news article and then conduct the same process to find the next source-indicative sentence. This iteration is repeated until the prediction probability of the model to the target source is below 10%.

By attributing the sentences of all news articles to news sources, we have the set of sentences (for each news source), which is crucial for source identification. To find the distinct sentences that appear only in a particular source, we reduce the dimension of the embedding vectors using the UMAP method (McInnes et al. 2018), and then group the sentences by using HDBSCAN. Note that we conduct this process for individual sources, so that a source may have multiple clusters. Finally, we identify 17.6 clusters for each source on average.

Analysis on Identified Patterns

Note that many source cues are irrelevant to political biases and hence should be ignored in political bias prediction. We itemize the significant patterns of sources cues as follows:

- **Whole Sentence:** Surprisingly, there are a few sentences whose texts appear in some of news articles of a particular source, which are likely to be captured either by the bias or source prediction model with a high probability. The example sentences are "*Newsletter Sign Up Continue reading the main story. Please verify you're not a robot by clicking the box.*" in `New York Times`, "*Get all the latest news on coronavirus and more delivered daily to your inbox.*" in `Fox News`, and "*You can unsubscribe at any time.*" in `Politico`.
- **Template:** The template sentences are also identified, which are the formatted texts used in several articles with different content texts. For example, the sentence "Update at {time}" frequently written in news articles in `NPR` while the first sentence in the most articles from `USA Today` tend to start with a city name (e.g., WASHINGTON).
- **Partial Sentence:** While the above cases are a form of a whole sentence, there is a case that a sentence contains one or more source-indicative signals. This is mostly related to the writing style of a news source such as specific words, phrases, or punctuation. For example, a pattern of two dashes ('–') frequently appear in the articles from `Fox news`. Similarly, capitalized preposition words such as 'On' or 'To' are often observed in `NPR` while the word 'Mr.' is distinctly used in `New York Times` and `Washington Times`.

Note that although the patterns of such sentences or templates can be excluded by simple processing, the prior work for political bias prediction has not considered removing these patterns. Furthermore, it is impossible to identify and remove all of partial-sentence source cues, implying that bias prediction models based on texts cannot completely avoid using these source cues.

4 Analyzing Dependence of Bias Prediction Model on Source Cues

In this section, we explore to what extent the prediction models for political bias depend on source cues. To this end, we first design six embedding models commonly used in prior work. We then evaluate (i) whether and how well these models can identify a news media of a given article and (ii) whether and how the decisions of these models change when we insert the source cues of news channels of different political bias.

Baseline Models

We select six embedding models with three different types as shown in Figure 1 as the target baseline models. For hierarchical models, we select HLSTM and HAN (Yang et al. 2016), which use the LSTM and the Transformer Encoder (TE) layers to extract the word- and sentence-level embeddings, respectively. These embedding models were imported

in the prediction models for political bias suggested by Li and Goldwasser (2019) and Ko et al. (2023)[1], respectively.

We also evaluate two baseline models, GNN and TE, which uses Gated-Relational Graph Convolutional Network (GNN) (Schlichtkrull et al. 2018) and TE (Lu et al. 2021) as sentence encoders, respectively, to generate an embedding of a news article based on the embeddings of individual sentences computed by the frozen RoBERTa. The GNN model was employed in (Zhang et al. 2022)[2] and (Feng et al. 2022).

Lastly, we also add BERT and RoBERTa as baseline models for the pre-trained type. Since both models are restricted by the length of input window (e.g., maximum 512 tokens), we use only the maximum length of the tokens from the beginning of a news article during evaluation. In all the experiments, we randomly split the dataset constructed in Section 3 as training and test sets with a 4:1 ratio. The technical details for implementation such as hyperparameters are discussed in Section B in the supplementary material.

Can Bias Prediction Models Predict the Sources of News Articles?

We first investigate whether and how accurately political bias prediction model estimate the source of a given article using the embeddings generated by the baseline models. To this end, we first train each baseline model to predict the political bias, and then replace the final decision layer of the baseline model into a new linear layer. We freeze all but the last layers and re-train the model to identify the news channel of a given article. The rationale behind this approach is that, if a model trained to predict the political bias of a news article relies heavily on the source information, the model can also identify the source of the news article.

Models	HLSTM	HAN	GNN	Trans-former	BERT	RoBERTa
Acc. (bias)	0.78	0.83	0.82	0.78	0.83	0.87
Acc. (source)	0.54	0.60	0.55	0.64	0.73	0.78

Table 2: Accuracy of bias-trained baseline models

Table 2 presents the results of experiments evaluating the baseline models for the source identification task. Note that each value is the average of running the experiment three times. The baseline models trained for political bias prediction can identify the source with more than 54% accuracy. Considering that we freeze the embedding layers of all the baseline models, this result demonstrates that the embeddings computed by the models for bias prediction reflect the source information, which implies that source information is used for political bias prediction.

Inserting Sources Cues To Incur Bias Flip

Here, we carry out fictitious experiments that try to flip the predicted bias by inserting the source cues (of a category) into a news article of another category, which is inspired by counterfactual generations (Garg et al. 2019) or adversarial attacks (Jin et al. 2020) in textual deep learning models. It turns out that inserting source cues into articles (from other news channels) can flip the prediction results, which could be evidence that the prediction model relies on the source cue in bias prediction.

Suppose we insert a source cue (e.g., "*View all New York Times letters.*") often found in articles from the `New York Times`, a left-leaning outlet, into an article of either center or right-bias. If the model is predicting the bias using the textual context, the model should ignore the inserted sentence. However, if the model uses the source cues in predicting the bias, the inserted sentence will lead the model to flip its bias prediction to left-bias.

The fictitious experiments are designed as follows. Note that this experiment is time-consuming since the inference time is proportional to (# of sources) $\times$ (# of injection sentences) $\times$ (# of target documents). Hence, we tried to reduce the number of trials. We truncate factors such as the number of sources, injection sentences, or target documents as follows. As for the number of sources, we use three largest sources for each bias, which are `Politico`, `Vox`, `New York Times` for *left-bias*, `NPR Online News`, `USA Today`, and `The Hill` for *center-bias*, and `Washington Times`, `Fox Online News` and `TownHall` for *right-bias*. Then, for each source, using the clusters of sentences in the previous section that share key patterns, we sample sentences from the cluster by the following criteria. (i) For **whole sentence** clusters which consist of a single sentence, we select these sentences to be inserted since it can be effective in predicting the source and also used by bias prediction models. (ii) In case of **template** sentence clusters, we randomly sample 10 sentences from each cluster. (iii) In case of **partial sentence** clusters, which contain patterns related to linguistic styles, we sample 20 expressions in each cluster. Note that we leave out small clusters (less than 30 sentences). Lastly, for the target document, among the 5,466 articles in the test set, we sample 500 articles for each insertion sentence.

After sampling the sentences and the target articles, we compare the bias prediction result of the original target article and the one with the inserted sentence. To minimize the change on the target document, rather than substituting one of the sentences in the article, we choose to prepend the sentence. Since news articles are often long documents, if the model correctly predicts the bias using the context or semantics, we expect the impact (i.e., bias flip) would be minimal.

Table 3 shows the statistics and results (i.e., average flip ratio of bias) of the experiments. Note that not all sentences are effective to flip the bias label; even the flip ratio of the sentences from the same cluster might differ significantly. Furthermore, even the same model trained multiple times might exhibit different behaviors in predicting biases depending on specific source-indicative patterns. Hence, for each source, we first classify "sampled sentences" to be inserted into two groups; ones from the **whole sentence** and **template** clusters (denoted by **S** in Table 3), and the others from **partial sentence** clusters (denoted by **P**). Note that

[1]https://github.com/yy-ko/khan-www23

[2]https://github.com/Wenqian-Zhang/KCD

	Politico		Vox		NY Times		NPR		USA Today		The Hill		WASH Times		Fox News		Townhall	
	S	P	S	P	S	P	S	P	S	P	S	P	S	P	S	P	S	P
# of all sentences	66	280	6	220	21	160	51	160	41	197	12	180	17	110	35	180	3	195
HLSTM	.58	.70	.43	.34	.32	.20	.68	.78	.27	.78	.11	.80	.14	.36	.41	.40	.33	.31
HAN	.41	.50	.23	.23	.19	.15	.13	.58	.32	.77	.06	.84	.23	.25	.40	.28	.44	.24
GNN	.78	.58	.71	.50	.53	.58	.61	.84	.54	.84	.16	.87	.28	.25	.46	.26	.47	.29
TE	.63	.24	.39	.25	.26	.11	.15	.51	.14	.34	.13	.76	.20	.21	.40	.16	.15	.16
BERT	.72	.83	.57	.67	.71	.59	.46	.87	.38	.91	.18	1.0	.15	.66	.43	.71	.84	.54
RoBERTa	.83	.82	.64	.75	.81	.50	.24	.65	.35	.93	.12	.93	.11	.29	.67	.54	.71	.48
AA	.12	.22	.10	.26	.10	.19	.30	.79	.55	.91	.07	.89	.25	.18	.40	.31	.50	.25
Rat-AA	.03	.13	.02	.14	.03	.11	.03	.17	.04	.17	.03	.16	.06	.17	.04	.18	.04	.16
Rat-AA + Reg.	.03	.12	.05	.13	.03	.08	.03	.17	.05	.19	.03	.14	.08	.24	.04	.22	.06	.23

Table 3: Bias flip rates of models in sentence insertion experiment is displayed. Each column stands for the media source that the cue is originated and the S and P in the following row are for the type of the source cue.

since the number of sentences in **P** is much larger than that of sentences that belong to **S**, we take into account only top 50% and 5% for **S** and **P**, respectively. Even though we average the flip ratios of successfully flipped cases, it can still prove that source-indicative information in a single sentence affects the model to flip the bias prediction.

From Table 3, we observe that inserting a single sentence causes a significant level of flipping biases across the six models. Though the flip rate varies by the source, the cue type, and the targeting model, every model can be substantially attacked by some of the source cues. For instance, we found that pre-trained language models (BERT, RoBERTa) are the most vulnerable whose flip rate is the highest across most news sources, with the average of 62% and 57%, respectively. Among the other models, we observe that sentence-level hierarchical model with the GNN has been affected by insertion most that it achieved 53% of average flip ratio in across sources. On the other hand, full hierarchical models (HLSTM and HAN) and sentence-level hierarchical (GNN and TE) models are less affected by the sentence insertion. Still it shows nearly 37% of bias flip on average.

While the whole sentence patterns (**S**) are the powerful cues in predicting the source, however, it has been revealed that the flip ratio differs by the sources. In case of sentences from `Politico`, it has achieved at least a 41% flip ratio (in HAN), while the sentences from `The Hill` shows significantly the lower flip ratio compared with other sentences of other news sources, or even partial sentence patterns of itself. We find that such results can be attributed to the number of sentences (of all the articles) in each source. For instance, a particular source cue in `Politico` appears in about 250 articles and often appears as groups of similar sentence patterns, while another source cue from `The Hill` appears in fewer articles (about 50).

Overall, by performing the bias flipping experiment, we found out that the six models are significantly affected by the textual patterns that could indicate their sources. Considering that (i) these models are trained to predict not sources but biases, (ii) bias prediction is affected by such source-indicative patterns, we believe that the models are predisposed to rely on source cues in predicting political biases.

5 A Model Minimizing Source Cues

From the previous two sections, we have observed that there are source-indicative patterns in the articles, and the models are prone to predict the bias using such patterns which have negative impact on the model's capability of predicting the bias. Thus, we propose a deep learning architecture that tries to avoid using the source-predictive patterns in predicting the bias.

Model Architecture

In (Baly et al. 2020), the authors have suggested to use Adversarial Adaptation (AA) (Ganin et al. 2016) to mitigate the influence of sources in predicting the bias, by proposing an architecture that adds an additional (news) source classifier at the output of the document encoder. By jointly minimizing both of the losses for bias prediction and source prediction, the gradient reversal layer between the source classifier and the document encoder will maximize the loss. Thus, the model learns how to process the document to predict its bias while independent of its source. However, we found that AA alone may not be enough in preventing the models from predicting the bias using the source cues, which motivates us to develop a model that explicitly prevents the source cues from being used for bias prediction.

In Figure 2, we describe the model architecture (RatAA) which mainly consists of three modules. The modules on the right side are the bias and source predictors. Each of these modules takes an article as an input and generates a document embedding (we use HAN as the document encoder), which is then passed to the linear classifier that produces the probability distributions for the bias and the source, respectively.

However, in the bias and source predictors, not all the sentences are given as the input. Since our motivation is to prevent the bias predictor from learning the source cues during training, we introduce the sentence selector in front of the

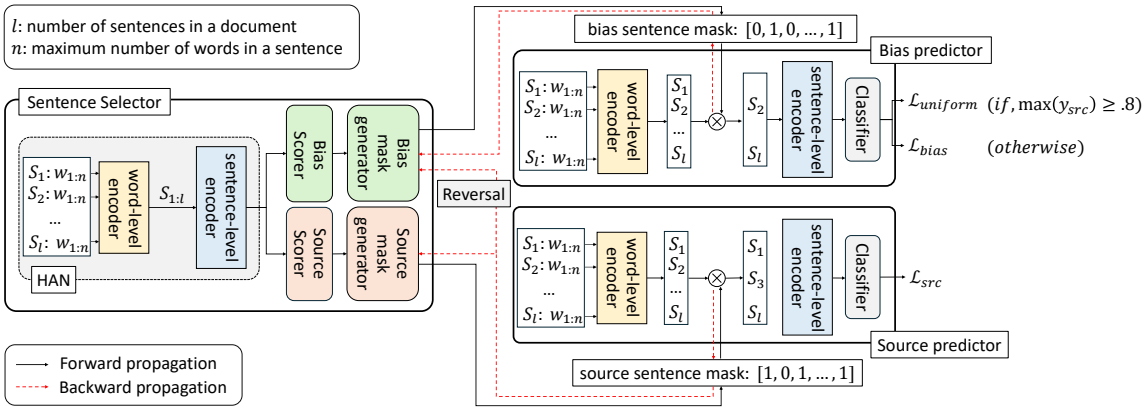


Figure 2: The overview of our model architecture, RatAA, is illustrated. (We omit the backward flow in each module.)

predictor, which passes only the subset of sentences from the input document to the predictors.

The idea of the sentence selector is from the rationalization architecture (Lei, Barzilay, and Jaakkola 2016), which is designed to find **only some parts of the input** (which is rationale) to make a prediction decision. In these studies (Lei, Barzilay, and Jaakkola 2016; Yue et al. 2023a), their architecture consists of two modules: (i) the generator that produces the rationale when given the input text, and (ii) the predictor that makes the prediction only with the given rationale. We take the similar approach; not all sentences are required in predicting the bias since (i) a substantial portion of a news article is factual, and (ii) source-indicative sentences should be removed. Thus, we combine the rationale framework with the AA mechanism to make the sentence selector learn the bias rationale, which are **bias-relevant** but **source-irrelevant** sentences.

The detailed mechanism of the sentence selector is as follows. During the forward propagation, a sentence (of a news article) is encoded as a sentence embedding, which is then processed through linear layers that output scores (for each sentence), which quantify how much the sentence will help predict the bias and the source, respectively. These scores are then given to the bias and source mask generators, respectively, each of which transforms the score into a binary mask whose value is 1 for the selected sentence, and 0 for those not.

The binary mask is then given as the input to each predictor where it is multiplied with the sentence embeddings $(S_1, S_2, ..., S_l)$, so that the predictor makes the prediction only using the selected sentences. Note that, during the backpropagation, we make the gradient from the source predictor flows to the bias mask generator with its values reversed, so that the gradient regarding the source prediction loss $(\mathcal{L}_{src})$ is given to the bias scorer in a reversed form. After combining the source prediction loss with the bias predictor loss $(\mathcal{L}_{bias})$, we expect the bias scorer to learn not only how to prioritize the sentences that are related to predicting the bias, but also how to penalize the sentences that are related to predicting the source.

To select sentences with high bias scores, we pass the scores through the bias mask generator which selects top $k_1\%$ of sentences which are expected to be bias-predictive but source irrelevant. A similar process is performed for the source prediction. Notice that we block top $k_2\%$ of the sentences since giving all the input sentences into the source predictor might result in finding the source patterns suboptimally. This is because the source predictor tends to stop finding any other patterns after it finds a small number of distinct patterns. To find as many source-indicative patterns as possible, we mask the high-score sentences, then pass only the remaining sentences to the source predictor, which then finds the other source patterns among the remainder.

In converting the sentence scores into the binary mask, we rely on Gumbel-softmax (Jang, Gu, and Poole 2017) during training. Gumbel-softmax is popular in rationale generation (Yue et al. 2023b). It enables end-to-end training, during which it adds some randomness in selecting the rationale which can be controlled by its temperature. During evaluation, we simply select top ranking sentences ordered by their scores.

While the sentence selector penalizes the source-cue sentences, it cannot prevent the bias predictor from learning such cues in the training phase. To further prevent the bias predictor learning from such source cues, we propose model with an additional regularization, RatAA+Reg, which regularize the bias predictor in the sense that, for source-indicative training samples, the model is driven not to predict the bias but to follow the uniform distribution $(\mathcal{L}_{uniform})$. In this paper, we apply the regularization mechanism as follows. When the bias mask is generated, we forward it to both the bias predictor and the source predictor. If the source predictor predicts a particular source with a probability higher than a pre-determined threshold (i.e., 80%), we conclude that the selected sentences contain a source cue.

Training: we design the training procedure as follows. First, we have pre-trained the sentence selector and the source predictor for a few epochs. If we train the bias predictor and the source predictor simultaneously, we found that the bias predictor converges faster than the source predictor. The sentence selector then fails to filter out the source cues. Second, while the baselines models were trained using the

early-stop with maximum patience of 10 epochs, we do not use the early stop in training this model. Since the higher accuracy on the test dataset might be the result of the model relying on the source cues, we do not use the test-set accuracy as criteria, but train the model for a predefined number of epochs.

We set the hyper-parameters as follows. For the training epochs, we have trained the sentence selector and the source encoder for the first 10 epochs, then we trained all modules for another 40 epochs. For the sentence selection ratio for the bias predictor (k_1), and the sentence mask ratio for the source predictor (k_2), we set 40%. In case of Gumbel-softmax, we set the temperature for the bias and source mask generators as 5.0 and 1.0, respectively. We set the temperature high for the bias mask generator, which allows the bias predictor to observe more sentences during training. As for the learning rate, the sentence selector is set to 1e-4, and two predictors are set to 2e-4.

Experimental Results

Here, we show the performance of the proposed models along with the model trained with AA. As for the bias prediction performance, we notice that the accuracy has declined from 80% (in case of baseline models) to nearly 65%. (62.7%, 67.6%, and 63% for AA, RatAA, RatAA+Reg, respectively) showing about 15% of drop in bias prediction. However, the high accuracy of the baseline models are due to strong reliance on the source-indicative cues. Thus, we evaluated the proposed models with the same experiments to find out whether they rely on the source cues or not. From the results in the bottom rows of Table 3, we confirm that by using AA, the models no longer rely on the source information since using AA only has achieved comparatively low flip rate with the sentence insertion.

Note that in the case of the center-bias sources, we find that the model with AA only still relies on the cues from those sources, with the flip rate reaching nearly 80%[3] On the other hand, both of our models (RatAA and RatAA with Regularization) achieve much lower flip rate across all sources indicating that our models successfully remove the reliance on source cues compared to the baseline models and the model with AA only.

Furthermore, we performed the experiments to find whether the document embedding generated in the proposed models contains the source predictable information. Hence, by training additional source classifier upon the frozen model, we found that the accuracy of the newly trained source predictor in the RatAA and RatAA+Reg is 0.45 and 0.4, respectively, indicating that these models substantially removed the influence of the source, compared with those of the baseline models (65% on average). Furthermore, the model with regularization (RatAA+Reg) has achieved lower source accuracy, which indicates the efficacy of the regularization scheme in removing the source information from the document embedding.

[3]We tested with different levels of adaptation (λ) and filled in the table with λ=0.8.

6 Discussions

Our experiments in Section 4 demonstrated that prior models for political bias prediction rely heavily on source cues. We contend that this outcome is, to some extent, inherent to the dataset. That is, the bias labels of news articles in the dataset are the same as those of their publishing sources. Thus, the models for political bias prediction tend to find source cues since they enable the models to accurately predict the political bias of a given article. An additional challenge in removing source cues is that these cues may contain relevant signals necessary for accurately predicting political bias, which could be important for improving model performance. However, the proposed model attempts to eliminate all source cues, regardless of whether they are relevant to bias, which may negatively impact overall bias prediction performance. While we made the contribution of addressing the disparity gap between source prediction and bias prediction, we clearly note the limitation of this paper. First, our evaluation for the prior models was performed with a public dataset and the results may vary for different datasets. However, since the labeling methods in the prior studies are almost identical, we believe that dependence on source cues will still exist. Second, there are a few recent studies using external knowledge in addition to the texts, which may reduce the dependence on source cues. Despite this, we believe that prior models still use the source cues as they compute the embeddings from the news texts.

7 Conclusion

In this work, we reveal that political bias prediction models trained using the available dataset in which news articles are labeled by their sources tend to be trained to predict the bias actually by predicting the source. Our experiment results show that prior models are prone to predict the bias by predicting the source of a given article using the source cues ranging from sentences to templates to other patterns, which are often irrelevant to its context/semantics. Thus, we propose a model whose central element is a sentence selector, which classifies the sentences (of an article) as either source-indicative ones and source-irrelevant ones. Note that only the latter ones are fed to the bias prediction module of the proposed model. Using such a filtering mechanism, it is shown that our model is resilient to fictitious settings where a source-cue sentence from a particular political stance (say, left-bias) inserted into a news article from a different news source (say, right-bias).

Acknowledgments

This work was supported by Korea government(MSIT) under the Institute of Information & communications Technology Planning & Evaluation (IITP) funded by the [NO.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University); IITP-2025-2021-0-02048; IITP-2025-RS-2024-00418784] and National Research Foundation of Korea (NRF). (No. RS-2023-00220985)

References

Auburn, L. 2023. Study of headlines shows media bias is growing. *University of Rochester*.

Baly, R.; Da San Martino, G.; Glass, J.; and Nakov, P. 2020. We Can Detect Your Bias: Predicting the Political Ideology of News Articles. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4982–4991. Online: Association for Computational Linguistics.

Barthel, M.; Mitchell, A.; Dorene Asare-Marfo, C. K.; and Worden, K. 2020. Measuring News Consumption in a Digital Era. Technical report, Pew Research Center.

Calice, M. N.; Bao, L.; Freiling, I.; Howell, E.; Xenos, M. A.; Yang, S.; Brossard, D.; Newman, T. P.; and Scheufele, D. A. 2023. Polarized platforms? How partisanship shapes perceptions of "algorithmic news bias". *New Media & Society*, 25(11): 2833–2854.

Cinelli, M.; Morales, G. D. F.; Galeazzi, A.; Quattrociocchi, W.; and Starnini, M. 2021. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9): e2023301118.

Dubois, E.; and Blank, G. 2018. The echo chamber is overstated: the moderating effect of political interest and diverse media. *Information, Communication & Society*, 21(5): 729–745.

Feng, S.; Chen, Z.; Zhang, W.; Li, Q.; Zheng, Q.; Chang, X.; and Luo, M. 2022. KGAP: Knowledge Graph Augmented Political Perspective Detection in News Media. arXiv:2108.03861.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; and Lempitsky, V. 2016. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17(59): 1–35.

Garg, S.; Perot, V.; Limtiaco, N.; Taly, A.; Chi, E. H.; and Beutel, A. 2019. Counterfactual Fairness in Text Classification through Robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, 219–226. New York, NY, USA: Association for Computing Machinery. ISBN 9781450363242.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*.

Jin, D.; Jin, Z.; Zhou, J. T.; and Szolovits, P. 2020. Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05): 8018–8025.

Kitchens, B.; Johnson, S. L.; and Gray, P. 2020. Understanding Echo Chambers and Filter Bubbles: The Impact of Social Media on Diversification and Partisan Shifts in News Consumption. *MIS Q.*, 44.

Ko, Y.; Ryu, S.; Han, S.; Jeon, Y.; Kim, J.; Park, S.; Han, K.; Tong, H.; and Kim, S.-W. 2023. KHAN: Knowledge-Aware Hierarchical Attention Networks for Accurate Political Stance Prediction. In *Proceedings of the ACM Web Conference 2023*, WWW '23, 1572–1583. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394161.

Lei, T.; Barzilay, R.; and Jaakkola, T. 2016. Rationalizing Neural Predictions. In Su, J.; Duh, K.; and Carreras, X., eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 107–117. Austin, Texas: Association for Computational Linguistics.

Li, C.; and Goldwasser, D. 2019. Encoding Social Information with Graph Convolutional Networks forPolitical Perspective Detection in News Media. In Korhonen, A.; Traum, D.; and Màrquez, L., eds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2594–2604. Florence, Italy: Association for Computational Linguistics.

Li, C.; and Goldwasser, D. 2021. Using Social and Linguistic Information to Adapt Pretrained Representations for Political Perspective Identification. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 4569–4579. Online: Association for Computational Linguistics.

Lu, J.; Henchion, M.; Bacher, I.; and Namee, B. M. 2021. A Sentence-Level Hierarchical BERT Model for Document Classification with Limited Labelled Data. In Soares, C.; and Torgo, L., eds., *Discovery Science*, 231–241. Cham: Springer International Publishing. ISBN 978-3-030-88942-5.

McInnes, L.; Healy, J.; Saul, N.; and Großberger, L. 2018. UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software*, 3(29): 861.

Pariser, E. 2011. *The filter bubble: What the Internet is hiding from you*. penguin UK.

Roscini, F. 2021. How The American Media Landscape is Polarizing the Country. *The Pardee Atlas Journal of Global Affairs*.

Schlichtkrull, M.; Kipf, T. N.; Bloem, P.; van den Berg, R.; Titov, I.; and Welling, M. 2018. Modeling Relational Data with Graph Convolutional Networks. In Gangemi, A.; Navigli, R.; Vidal, M.-E.; Hitzler, P.; Troncy, R.; Hollink, L.; Tordai, A.; and Alam, M., eds., *The Semantic Web*, 593–607. Cham: Springer International Publishing. ISBN 978-3-319-93417-4.

Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; and Hovy, E. 2016. Hierarchical Attention Networks for Document Classification. In Knight, K.; Nenkova, A.; and Rambow, O., eds., *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–1489. San Diego, California: Association for Computational Linguistics.

Yue, L.; Liu, Q.; Wang, L.; An, Y.; Du, Y.; and Huang, Z. 2023a. Interventional Rationalization. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 11404–11418. Singapore: Association for Computational Linguistics.

Yue, L.; Liu, Q.; Wang, L.; An, Y.; Du, Y.; and Huang, Z. 2023b. Interventional Rationalization. In Bouamor, H.;

Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 11404–11418. Singapore: Association for Computational Linguistics.

Zhang, W.; Feng, S.; Chen, Z.; Lei, Z.; Li, J.; and Luo, M. 2022. KCD: Knowledge Walks and Textual Cues Enhanced Political Perspective Detection in News Media. In Carpuat, M.; de Marneffe, M.-C.; and Meza Ruiz, I. V., eds., *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4129–4140. Seattle, United States: Association for Computational Linguistics.