CBS COPENHAGEN BUSINESS SCHOOL
HANDELSHØJSKOLEN

Can political bias in news articles be accurately detected using NLP techniques?

Natural Language Processing and Text Analytics (CDSCO1002E)

Student: Johannes Stärk

Pages: 15

Characters (with Space): 31.560

Date: May 30, 2025

Link to data & other files:
https://drive.google.com/drive/folders/1bmvcajpeBadzhCcjOfKsr4IAMazvthgF?usp=sharing

Copenhagen Business School
Department of Digitalization
2000 Frederiksberg
rg.digi@cbs.dk

Abstract

This paper examines the effectiveness of NLP-based classification models in detecting political bias in news articles. A variety of approaches are evaluated, ranging from classical NLP models such as Naive Bayes, which use sparse lexical features, to advanced transformer-based models with contextual embeddings. Our experiments demonstrate that political bias can be reliably identified, with performance depending heavily on the complexity of feature representations and model architectures. Logistic Regression with TF-IDF achieves solid results (an accuracy of 0.69) and serves as a baseline model. Although combining dense embeddings with neural models offers limited improvement, transformer models such as RoBERTa achieve the highest accuracy (0.87) by capturing nuanced contextual and stylistic patterns. However, these gains come at the cost of substantial computational resources. Overall, our results highlight the clear trade-off between model performance and resource efficiency. This suggests that classical models remain a viable option when computational constraints or interpretability are priorities.

Keywords: Natural Language Processing - Text Classification - Political Bias Detection - Transformer Models

Contents

1 Introduction

In recent years, digital news consumption has become the primary way people stay informed. According to a study, over 86% of U.S. adults access news through digital devices, with more than half (54%) encountering it via social media platforms (Pew Research Center, 2024). While this makes news more accessible, it also introduces new challenges. Algorithms tend to recommend content similar to what users have previously engaged with, which can reinforce existing beliefs and filter out opposing viewpoints (Groshek & Koc-Michalska, 2017). Furthermore, social media platforms do not always clearly identify the source or credibility of news content, which increases the risk of misinformation and bias .

As the media landscape becomes more ideologically divided, individuals are increasingly likely to be exposed primarily to viewpoints that align with their own. This can lead to "information bubbles", where readers may be unaware of how linguistic framing influences their interpretation. Political bias in news reporting, whether intentional or not, plays a role in shaping public opinion and reinforcing political divisions (Rodilosso, 2024).

Therefore, understanding and identifying political bias in news is a relevant and pressing challenge. Methods that can automatically detect such bias could contribute to efforts aimed at improving media transparency, supporting critical thinking, and promoting a more informed and balanced public discourse. This work focuses on classifying news articles to detect political bias using natural language patterns.

Business & Social Relevance

In today's crowded digital news landscape, platforms and publishers must strike a balance between engaging users and maintaining credibility. Automated bias detection technologies that can classify articles as left-wing, centre-wing or right-wing enable media companies to present a more balanced mix of content. This improved curation can strengthen audience trust and support advertising strategies by providing clearer audience segmentation based on the political leaning of the content. Furthermore, organisations can monetise bias insights by licensing analytics services, offering subscription-based reporting tools, or integrating with existing content management systems.

Beyond commercial applications, bias detection can also contribute to social well-being. Real-time identification of ideological biases enhances media literacy by making framing techniques transparent and encouraging readers to seek out diverse viewpoints (Pennycook & Rand, 2019). In an era of increasing polarisation, such tools can help to revive media pluralism and encourage more informed public discourse.

2 Related Work & Research Questions

A growing body of work has explored how to classify political bias in news articles, using a variety of input representations and model architectures.

Nadeem and Raza conducted a comprehensive evaluation of classical and neural models on article- and sentence-level bias detection. They found that count-based features such as Bag-of-Words and TF-IDF are poorly suited to detecting political bias, primarily because these approaches treat each word independently and ignore syntax and semantics. Their analysis showed that left-, center-, and right-leaning articles often use similar vocabulary, especially when reporting on shared topics like government policy or elections. As a result, models based solely on surface-level lexical differences tend to misclassify articles that use ideologically ambiguous or balanced language. Even attempts to apply clustering or dimensionality reduction on these features failed to produce clear class separability, suggesting that political bias cannot be inferred from word frequency alone, but rather depends on contextual framing and the subtle connotations of word usage (Nadeem & Raza, 2023).

Jin and Yin evaluated both standard and hierarchical transformer-based models on the SemEval-2019 Hyperpartisan News Detection dataset. Their experiments demonstrated that a fine-tuned BERT classifier achieved exceptionally high accuracy (up to 97%) on five-class source-labeled data. Interestingly, a more complex architecture (combining BERT with a Hierarchical Attention Transformer) did not outperform the simpler fine-tuned BERT model. This suggests that when articles are short or homogeneous in structure, flat transformer models may already capture the key ideological cues. Their results highlight that contextual embeddings from fine-tuned transformers remain the most effective representation for bias detection, even compared to more structured or multi-stage models (Jin & Yin, 2023).

Liu et al., the authors of the dataset used in our work, emphasized similar concerns. In their introduction of the BIGNEWS corpus, they showed that while transformer-based classifiers perform well, this may be partly due to the presence of publisher-specific language patterns. If models are not carefully controlled for label leakage, such as by removing source metadata, they risk overfitting to stylistic cues rather than ideological content. Their work underscores the importance of focusing on article-level text alone when attempting to measure bias in a way that generalizes across sources (Liu et al., 2022).

Building on these insights, our paper aims to compare a range of feature representations and model types to evaluate their effectiveness in detecting political bias in news articles. While prior work indicates that contextualized models tend to perform better, there is still limited comparative evidence on how different representations perform relative to each other. Our paper aims to shed light on this by benchmarking a range of approaches

on the same dataset and task. This leads us to the following research questions:

- **Main Research Question:** Can we use NLP techniques to accurately detect political bias in news articles?
- **Sub-Questions:**
  - **Feature Complexity vs. Accuracy:** How much does classification accuracy improve as we progress from simple count-based features to fully fine-tuned transformer embeddings?
  - **Computational costs and interpretability vs. accuracy :** What are the trade-offs between accuracy, computational costs and interpretability?

3 Conceptual Framework

To narrow down the broader challenge of identifying political bias in digital news, we frame the task as a supervised classification problem. The objective is to assign each article to one of three political orientations: left, center, or right. This approach is grounded in the assumption that political leanings are reflected in language patterns, including word usage and sentence structure.
Our framework (see Figure 1) begins with sampling and splitting the dataset into training and test sets. Prior to modeling, the text undergoes various levels of preprocessing depending on the representation method, ranging from no or minimal preprocessing to more extensive transformations.
The processed text is then converted into numerical representations of each news article using several feature extraction techniques:

- **Sparse vectors** based on Bag-of-Words and TF-IDF
- **Averaged dense vectors** using Word2Vec embeddings
- **Contextual embeddings** from a pre-trained SBERT model
- **Fine-tuned CLS vectors** from transformers (BERT and RoBERTa)

These different approaches enable us to explore which types of language signals (lexical, syntactic, or contextual) are most indicative of ideological stance.
The feature vectors are used to train and evaluate a range of models. We apply both classical machine learning algorithms, such as logistic regression and neural networks, and more advanced transformer-based classifiers. Finally, we evaluate and compare multiple combinations of input features and models.

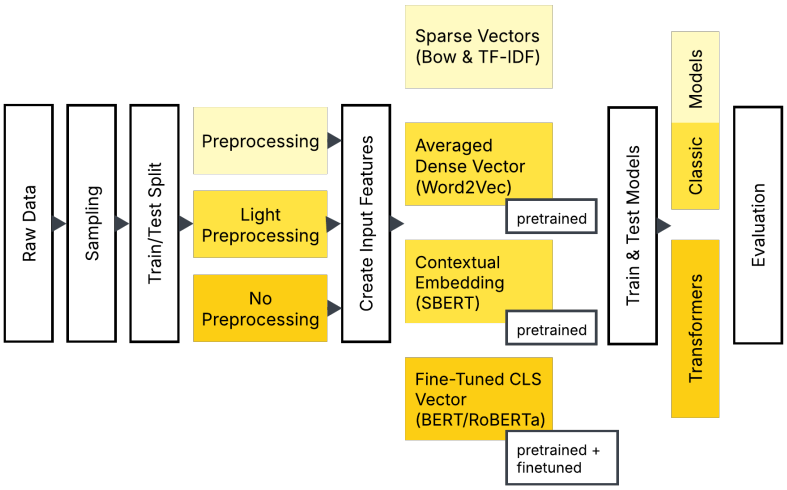

Figure 1: Conceptual Framework

4 Data Set Description & Analysis

We use the BIGNEWSBLN dataset, a balanced and preprocessed subset of the larger BIGNEWS corpus introduced by Liu et al. It consists of approximately 2.33 million news articles from 11 major U.S. media outlets, each categorized as left, center, or right. The assignment of the label is based on ideological ratings from AllSides and Ad Fontes Media. Importantly, the bias label reflects the outlet's political leaning, not a human annotation of each individual article (assuming uniform political bias within each publisher's output) (Liu et al., 2022).

To ensure ideological balance, the dataset was downsampled so that each political category is equally represented. It includes only U.S. domestic political news, filtered to retain articles from relevant categories such as elections, government policy, and political figures.

The dataset includes the following four columns: (news)title, (news)text, source and the (bias)label ("left", "center" or "right"). For the purposes of model training and evaluation, we rely exclusively on the text and bias label columns. As the label is derived from the publisher, the source column is excluded as it would leak the label. Additionally, we do not include the title in our modeling pipeline, as headlines often follow formulaic or click-oriented structures that may not fully reflect the article's linguistic framing or substantive content.

Looking at the distribution of publishers in the dataset, notable asymmetries across political categories can be seen (See Fig. 2). Right- and center-leaning articles are dominated by a few high-volume publishers: Fox News and Breitbart for the right, and The Hill and Associated Press for the center. In contrast, left-leaning articles are more evenly

distributed across a larger number of smaller publishers. This uneven concentration implies that articles labeled as left come from a more stylistically diverse set of sources, whereas right and center articles may be more homogeneous in tone and format due to the dominance of a few outlets.

Furthermore, most articles in our dataset are relatively short: the majority are concentrated between 100 and 400 words (see Fig. 3). This is particularly relevant given that transformer-based models such as BERT and RoBERTa typically operate on inputs with a maximum length of 512 tokens. Thus, most articles can be processed in full without truncation, minimizing potential information loss. However, despite the manageable length of individual articles, the overall scale of the dataset poses a computational challenge. With an average of roughly 300 words per article across 2.2 million entries, the corpus amounts to approximately 660 million words, suggesting a sampling strategy.

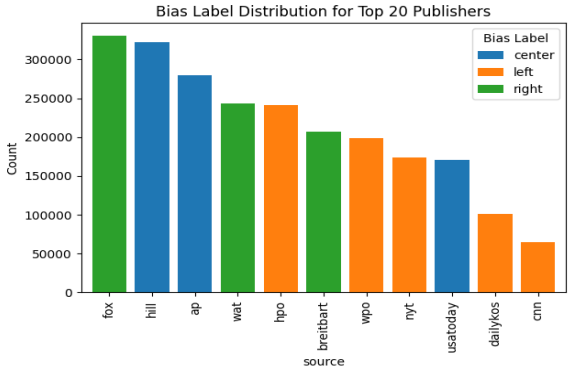

Figure 2: Distribution of Publishers

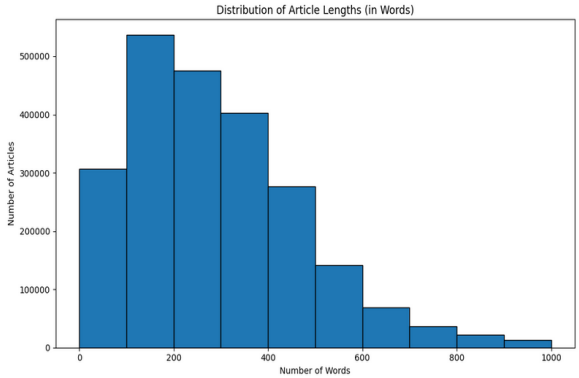

Figure 3: Distribution of Article Lengths

To explore lexical patterns in our dataset, we analyzed the 50 most frequent words for each political class. The results reveal substantial overlap in vocabulary across left-, center-, and right-leaning articles. Words like "said", "president", "new", "state", "trump", and "government" consistently rank among the most common in all three categories.

While some slight differences in word usage exist, for example, left-leaning texts include more mentions of "political", or "party", and right-leaning articles include words like "news" or "wallace"; these variations are relatively minor in the context of overall vocabulary use.

This observation highlights the challenge of distinguishing political bias at the surface level of word frequency and suggests that more subtle language patterns may underlie ideological differences.

5 Experimental Setup

5.1 Data Preparation

Given computational limitations, we sampled a balanced subset of 150,000 articles from the full BIGNEWSBLN dataset, drawing 50,000 randomly from each political bias category (left, center, right). To ensure reproducibility, a fixed random state was used during sampling. Articles that explicitly referenced their publisher in the text were removed to reduce label leakage.
We applied three levels of preprocessing on article texts, depending on the embedding method used:

- **Full preprocessing for sparse vectors (Bag-of-Words and TF-IDF):** Tokenization, lowercasing, and removal of punctuation and stopwords were applied. These steps help reduce feature space noise and sparsity, improving the quality and interpretability of frequency-based text representations.
- **Light preprocessing for dense embeddings (e.g., Word2Vec):** Only tokenization and lowercasing were performed. Stopwords and punctuation were retained, as they can carry valuable contextual cues that aid in capturing word relationships within dense embedding spaces.
- **No preprocessing for transformer-based embeddings (e.g., BERT, RoBERTa):** These models include their own tokenization pipelines and are pretrained on raw text. Applying additional preprocessing could interfere with their internal structure and diminish performance.

The processed dataset was then split into training and test sets to support evaluation on unseen data.

5.2 Feature Extraction Methods and Model Selection

To systematically evaluate the trade-off between performance and complexity, we adopted a progressive approach to feature extraction and model selection. Starting with simple, interpretable methods such as Bag-of-Words, we incrementally introduced more sophisticated and resource-intensive techniques, including contextualized embeddings from transformer models. This allowed us to track improvements in classification accuracy while also considering computational efficiency and feasibility.
Input features:

- **Bag-of-Words & TF-IDF**: Despite prior work indicating that count-based features are limited in capturing the syntactic and semantic nuances of political bias,

we include them in our experiments as baseline models to benchmark the performance of more advanced embedding-based approaches.

- **Averaged Word Embeddings (Word2Vec)**: Articles were encoded by averaging pre-trained word vectors from the Google News corpus. This approach captures basic semantic similarity between words but does not consider word order or syntax.
- **Sentence Embeddings (SBERT)**: Modeled sentence-level meaning using a pre-trained transformer. Each document was encoded into a single vector by applying mean pooling over all token embeddings, preserving overall semantic context.
- **Fine-Tuned Transformer Embeddings (RoBERTa/BERT)**: Entire transformer models were fine-tuned on the classification task. Final representations were derived from the `[CLS]` token, allowing the model to learn task-specific contextual embeddings in an end-to-end manner.

Model selection followed a similarly progressive approach:

- **Classical Models (Naïve Bayes, Logistic Regression)**: Provided fast baselines and interpretable results.
- **Intermediate Neural Models (MLP, BiLSTM)**: Allowed modeling of non-linear patterns and sequential data.
- **Transformer-Based Architectures (Fine-tuned BERT, RoBERTa)**: Leveraged deep contextual encoding and end-to-end optimization.

We hypothesized that combining more sophisticated embeddings with higher-capacity models would better capture the subtle linguistic indicators of political bias and ultimately yield higher classification performance.

5.3 Evaluation Metrics

We evaluated model performance comprehensively using four metrics: accuracy, precision, recall, and F1-score. Although the dataset was intentionally balanced, using precision, recall, and F1-score alongside accuracy provided detailed insights into each model's classification behavior. This ensured a thorough evaluation of the models' strengths, weaknesses, and overall ability to detect political bias across all categories.

6 Performance Evaluation & Interpretation

6.1 Baseline Models with Sparse Embeddings

Model	Features	Accuracy	Precision			Recall			F1-Score		
			Left	Center	Right	Left	Center	Right	Left	Center	Right
1. Naive Bayes	Bag of Words	0.55	0.62	0.54	0.52	0.51	0.59	0.56	0.55	0.56	0.53
2. Naive Bayes	TF-IDF	0.56	0.62	0.55	0.54	0.54	0.58	0.56	0.57	0.56	0.56
3. Logistic Regression	Bag of Words	0.68	0.68	0.67	0.68	0.70	0.67	0.66	0.69	0.67	0.67
4. Logistic Regression	TF-IDF	0.69	0.70	0.67	0.70	0.70	0.70	0.66	0.70	0.68	0.68

Table 1: Baseline model performance using sparse vectors (BoW & TF-IDF).

We began our experiments with two simple classification models: Naive Bayes and Logistic Regression, each tested with Bag-of-Words and TF-IDF features. Naive Bayes achieved relatively modest results, with accuracies of 0.55 using Bag-of-Words and 0.56 using TF-IDF. This performance likely reflects the limitations of its core assumption of feature independence, which does not align well with the nature of political news articles. In this context, subtle patterns of co-occurring words and framing expressions often signal bias. For instance, the difference between phrases like "radical policy" and "progressive policy" can be significant, yet Naive Bayes lacks the ability to capture such dependencies. Logistic Regression, on the other hand, performed surprisingly well, slightly beyond our initial expectations. It reached 0.68 accuracy with Bag-of-Words and 0.69 with TF-IDF. In addition to strong overall accuracy, its precision, recall, and F1-scores were stable across all three bias categories, each hovering around 0.67 to 0.70. This balanced performance suggests that the model was not favoring any one class disproportionately and was able to generalize reasonably well across the ideological spectrum.

Bias Label	Avg. Nouns		Avg. Adjectives		Avg. Length	
	Misclassified	Correctly Classified	Misclassified	Correctly Classified	Misclassified	Correctly Classified
Left	126.30	166.15	34.71	50.62	306.75	396.58
Center	112.42	127.69	31.54	33.15	270.41	298.60
Right	119.38	133.16	33.35	37.37	289.37	339.28

Table 2: Comparison of average linguistic features between misclassified and correctly classified articles, grouped by bias label.

To better understand how linguistic features relate to classification performance, we applied part-of-speech tagging and compared misclassified and correctly classified articles from our best baseline model (logistic regression using TF-IDF). We focused on the number of nouns, adjectives, and article length.

As shown in Table 2, correctly classified articles consistently contain more nouns and adjectives and are longer on average across all bias categories. This pattern is expected,

especially for a TF-IDF-based model, which relies on term frequencies to identify relevant features. Articles that are richer in descriptive content provide more distinctive and varied terms, giving the model more information to work with. In contrast, shorter or less content-dense articles may contain fewer informative or unique words, making it harder for the model to identify meaningful patterns and leading to more misclassifications.
In summary, despite prior work showing that frequency-based features like Bag-of-Words and TF-IDF are not well-suited for detecting political bias, both performed equally well in our experiments. We had expected TF-IDF to outperform due to its term-weighting, but the difference was minimal. This suggests that even simple models can provide strong baselines for political bias detection.

6.2 Neural Networks with Dense Embeddings

Model	**Features**	**Accuracy**	**Precision**			**Recall**			**F1-Score**		
			Left	Center	Right	Left	Center	Right	Left	Center	Right
5. MLP	Word2Vec	0.59	0.59	0.58	0.58	0.65	0.50	0.53	0.61	0.54	0.55
6. BiLSTM	Word2Vec	0.63	0.64	0.59	0.67	0.63	0.69	0.57	0.63	0.64	0.61
7. MLP	SBERT	0.52	0.54	0.52	0.54	0.51	0.53	0.52	0.52	0.52	0.53
Comparison with Baselines											
8. Logistic Regression	Word2Vec	0.54	0.57	0.54	0.51	0.60	0.54	0.48	0.58	0.54	0.50
9. Logistic Regression	SBERT	0.53	0.55	0.53	0.51	0.54	0.52	0.51	0.54	0.53	0.51
10. Logistic Regression	Chunked SBERT	0.57	0.59	0.57	0.54	0.60	0.58	0.52	0.59	0.57	0.53

Table 3: Model performance using dense word or sentence embeddings.

To explore whether more expressive feature representations could enhance classification performance, we experimented with two dense embedding methods: static Word2Vec and contextual SBERT, combined with neural network models such as MLPs and a BiLSTM. Dense embeddings are well-suited for nonlinear classifiers, as they can capture subtle patterns and deeper semantic relationships within the text.

MLP and BiLSTM with Word2Vec
Both models were built on the same frozen 300-dimensional Word2Vec embeddings. The MLP used a simple feedforward design that converged reliably under Adam optimization with early stopping. To avoid suboptimal configurations, we conducted extensive grid searches for each model, tuning the sizes of the hidden layers, the activation functions, the regularisation strengths, and the learning rates. Despite this, the MLP achieved an accuracy of only 0.59, performing poorly compared to our strongest baseline.
Believing that word order and context could improve the results, we implemented a compact BiLSTM model. After tokenisation and padding, the embeddings were passed through Gaussian noise and batch normalisation for regularisation. The BiLSTM layer comprises 32 units in each direction with 50% recurrent dropout. This is followed by a

32-unit dense layer with SELU activation and L2 decay ($\lambda = 0.01$), and a final softmax output layer. Training was performed using Adam (lr = 0.001) with early stopping based on validation loss (patience = 3). On the test set, the BiLSTM achieved an accuracy of 0.63, outperforming the MLP by leveraging sequential dependencies and local context patterns. This advantage is likely due to the BiLSTM's ability to capture word order and temporal relationships in the text.

These relatively modest results prompted us to examine Word2Vec's representational capacity for ideological signals more closely. First, we selected the most frequent words unique to each class and projected their Word2Vec representations into two dimensions using PCA. As shown in Figure 4, there are some signs of grouping, suggesting that Word2Vec captures certain ideological tendencies. However, the classes are not well separated in the embedding space, and many points remain intermixed.

We also visualized embeddings for a curated list of framing words derived from D'Alonzo and Tegmark, which are known to reflect ideological language (D'Alonzo & Tegmark, 2021). As shown in Figure 5, the overall separation between left- and right-leaning terms remains weak for these specific words as well. This suggests that while Word2Vec captures some semantic distinctions, it may not offer strong enough signals to support robust political bias classification, especially in models that depend on high-quality vector representations.

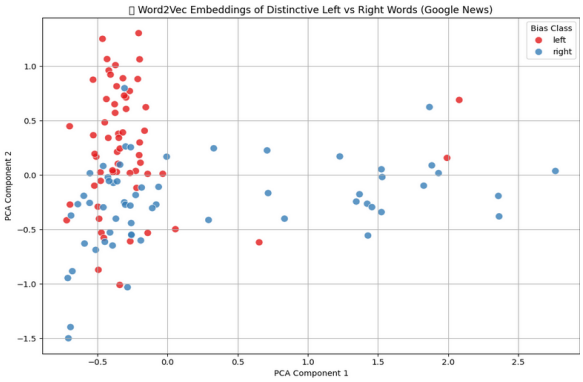

Figure 4: PCA visualization of Word2Vec embeddings

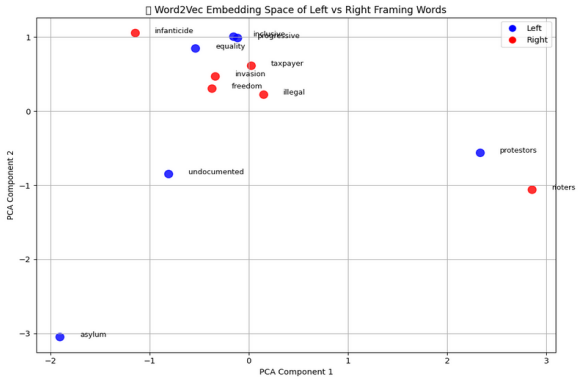

Figure 5: Distribution of Article Lengths

SBERT Sentence Embeddings

To leverage contextual, sentence-level information, we substituted Word2Vec with SBERT encodings next. Initially, we averaged all the sentence embeddings to create a single vector for each article and trained an MLP. However, this approach only achieved an accuracy of 0.52, indicating that simple pooling tends to obscure key distinctions between sentences. Seeking a richer structure, we implemented a 'chunked' method: each article was split into up to 20 sentences, each of which was encoded individually and then averaged with the others. Despite using a powerful SBERT variant (all-mpnet-base-v2), this more gran-

ular pipeline did not improve performance. These results suggest that even sophisticated mean pooling of SBERT vectors may discard the critical inter-sentence dynamics needed to distinguish nuanced ideological framing in our three-way bias task. This led us to discontinue further SBERT-based experiments.

Comparison to Baselines

Our optimised BiLSTM using Word2Vec outperforms Logistic Regression with the same embeddings, confirming the value of sequential modeling. However, both neural models underperform compared to the strongest baseline (Logistic Regression with TF-IDF), which achieved 0.69 accuracy. Even Logistic Regression with SBERT embeddings (0.53 accuracy) performed worse than TF-IDF, underscoring the challenges of using dense, pretrained embeddings in this context. Word2Vec assigns one vector per word, ignoring context, while SBERT may not align well with political discourse due to pretraining on general-purpose data. These limitations highlight why, despite their simplicity, count-based features remain highly competitive for political bias detection under real-world constraints.

6.3 Transformers

Model	**Features**	**Accuracy**	**Precision**			**Recall**			**F1-Score**		
			Left	Center	Right	Left	Center	Right	Left	Center	Right
11. RoBERTa	Transformer Embeddings	0.87	0.84	0.89	0.89	0.89	0.87	0.86	0.87	0.88	0.87
12. BERT	Transformer Embeddings	0.86	0.83	0.88	0.88	0.89	0.86	0.85	0.86	0.87	0.86

Table 4: Performance of fine-tuned transformer-based models.

As expected, transformer models like BERT and RoBERTa achieved the best results in our paper. Unlike simpler methods such as Bag-of-Words, TF-IDF, or static embeddings like Word2Vec and SBERT, these models generate contextualized representations that reflect how words are used in different contexts. Because both models were fine-tuned on our classification task, they could learn the specific language patterns that signal political bias. Both models clearly outperformed earlier approaches. As shown in Table 4, RoBERTa reached an accuracy of 0.87, slightly higher than BERT's 0.86. This is substantially above the 0.69 accuracy of our best classical approach, Logistic Regression with TF-IDF, and the 0.63 accuracy of our top dense-embedding model, BiLSTM with Word2Vec. Performance was consistently high across all three bias categories (left, center, and right), demonstrating balanced detection without favoring any single class.

RoBERTa builds on BERT's architecture but incorporates a larger pretraining corpus, removes the Next Sentence Prediction objective, and uses dynamic masking. We expected these enhancements to give RoBERTa an edge over BERT, and indeed it slightly outperformed BERT (Kacprzak, 2025). However, this improvement is marginal and would be

negligible for most practical applications, suggesting that BERT remains a competitive choice when computational resources are limited.

7 Discussion

Our experiments confirm that NLP techniques can reliably detect political bias in news texts. Fine-tuned transformers produced the best results, with BERT achieving 86% accuracy and RoBERTa achieving 87%. Both models maintained nearly identical precision, recall and F1-scores (approximately 0.86–0.89) across the left, center and right categories. These figures demonstrate the stability and robustness of deep contextual embeddings for three-way bias classification. Overall, our findings validate the feasibility and accuracy of automated bias detection using state-of-the-art transformer models.

Feature Complexity vs. Accuracy

Figure 6 illustrates this relationship by plotting accuracy against a combined feature and model complexity scale. Each point represents a model-feature combination. Lower complexity methods (Bag-of-Words and TF-IDF with Naïve Bayes or Logistic Regression) are plotted on the left, while higher complexity approaches (embeddings and transformers) are plotted on the right. The clear upward trend reflects the positive overall correlation: as text representation richness and model sophistication increase, so does classification accuracy.

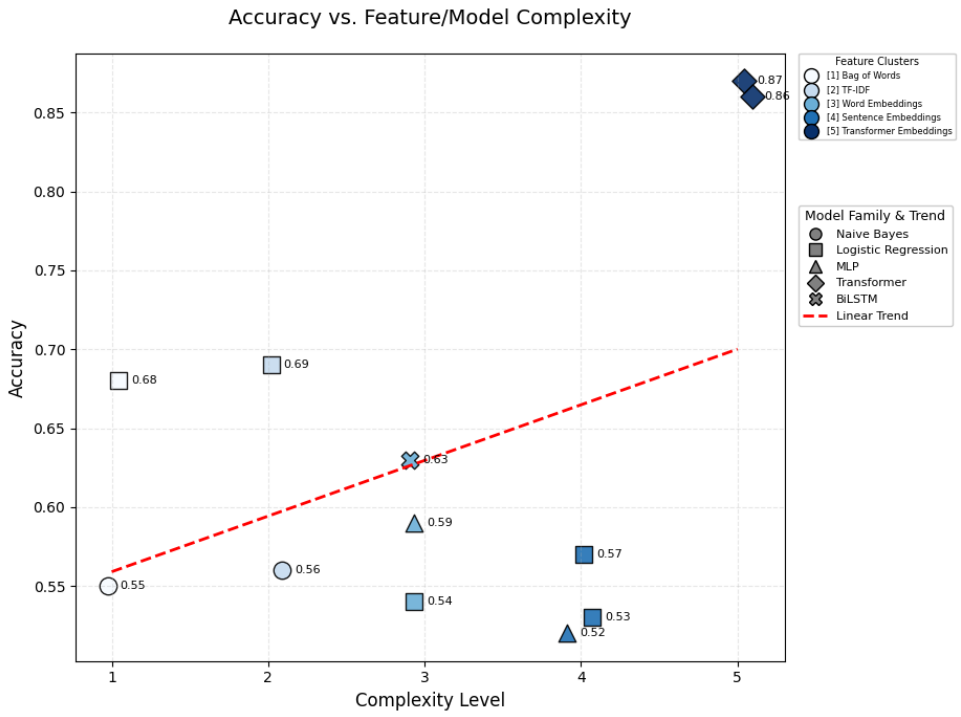

Figure 6: Relationship accuracy vs. combined feature and model complexity

However, the plot also shows that simpler methods, especially Logistic Regression with TF-IDF, already capture a large proportion of bias signals, achieving almost 70% accuracy. Mid-level embeddings, such as SBERT, offer modest improvements of around 53–57%, while shallow neural models based on Word2Vec achieve accuracies of around 59–63%. The most substantial improvement occurs at the transformer level, with accuracies surpassing 86%. This suggests that, although advanced models offer the best performance, simpler techniques remain highly competitive and more efficient in terms of resources.

Computational costs and interpretability vs. accuracy

To better interpret these results, it is useful to consider how performance scales with model complexity and what trade-offs arise in terms of interpretability and computational cost. Our findings highlight a clear trade-off between interpretability, computational costs, and performance. Simpler models using count-based features, offer high interpretability and very low computational cost. These models allow for direct inspection of influential words, can be trained within few minutes on CPU, and yield solid performance with an accuracy of around 0.69%.

Models based on static or sentence-level embeddings, such as Word2Vec or SBERT paired with shallow neural networks, introduce more sophisticated representations of textual meaning and modestly increase computational requirements. However, their interpretability is lower, as the internal structure of embeddings and hidden layers obscures the direct relationship between input features and predictions. Despite this added complexity, these models offer only moderate performance improvements, with accuracies ranging between 0.52% and 0.63%.

The most advanced techniques, fine-tuned transformer models like BERT and RoBERTa come with substantial computational demands, both during training and inference, which we're only feasible with use of GPU. Moreover, they behave largely as black boxes, offering limited interpretability without the use of specialized tools.

7.1 Limitations & Ethical Considerations

Our results demonstrate the feasibility of using NLP models to classify political bias. However, this task is subject to inherent methodological and ethical limitations.

The primary limitation lies in the nature of the ground truth. The bias labels are derived from the publisher's general ideological stance rather than the specific linguistic or argumentative content of individual articles. For example, a left-leaning editorial published on a right-leaning platform would still be labelled as 'right', introducing label noise. In this context, models may also learn to identify characteristics of the publisher, such as style, tone, or formatting, instead of actual ideological bias. This highlights the broader

issue of subjectivity in political labelling. Without a universally accepted framework for ideological categorisation, achieving 100% accuracy still only reflects the dataset's internal labelling logic. Political bias is often expressed through subtle discourse rather than specific vocabulary, making it difficult to detect. While models like BERT and RoBERTa capture semantic nuance better than sparse methods, they still face challenges with context-dependent features such as irony or rhetorical framing (Maladry et al., 2023). Shared vocabulary across political categories further complicates this task.

These modelling limitations have implications beyond technical performance, including ethical concerns related to fairness and potential real-world consequences. Misclassification can distort public perception and damage authors' reputations, particularly in polarised environments.

Simplified labels may also reinforce filter bubbles. Without an understanding of the context, consumers of media may rely on such labels as proxies for credibility or bias, thereby deepening partisan divides. To mitigate these risks, transparency and interpretability are essential. Classification systems should be accompanied by explanations that clarify the basis for predictions, particularly in applications that could influence public discourse.

8 Conclusion & Future Work

Our findings confirm the main research question: Political bias in news articles can be reliably detected using NLP-based classification models. Regarding the first sub-question, we observe that the complexity of features and models clearly impacts performance. Logistic Regression with TF-IDF achieves solid accuracy of 0.69, while transformer-based models like RoBERTa can reach 0.87 by effectively capturing subtle linguistic and contextual cues. However, simpler models with sparse representations still offer competitive and interpretable results. In relation to the second sub-question, we find that these performance improvements incur significantly higher computational costs. Advanced models require substantial resources and training time, whereas classical approaches remain efficient and practical in scenarios focused on low resources or interpretability.

Our paper identifies three areas for future research. Firstly, shifting the focus from publisher-level labels to article-level classification can decrease label noise. Secondly, as bias often emerges through framing rather than individual words, incorporating discourse-level features such as argument structure and narrative flow may reveal hidden ideological implications. Thirdly, visualising attention weights or key phrases would enhance the transparency of explainable AI and foster greater accountability in bias detection tools.

References

D'Alonzo, S., & Tegmark, M. (2021). Machine-Learning media bias [Publisher: arXiv Version Number: 1]. https://doi.org/10.48550/ARXIV.2109.00024

Groshek, J., & Koc-Michalska, K. (2017). Helping populism win? Social media use, filter bubbles, and support for populist presidential candidates in the 2016 US election campaign. *Information, Communication & Society*, *20*(9), 1389–1407. https://doi.org/10.1080/1369118X.2017.1329334

Jin, E., & Yin, E. (2023). Classifying Partisan Bias in News Articles: Leveraging an Understanding of Political Language and Article Structure. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1234/final-reports/final-report-169502805.pdf

Kacprzak, K. (2025). RoBERTa vs. BERT: Exploring the Evolution of Transformer Models. https://www.dsstream.com/post/roberta-vs-bert-exploring-the-evolution-of-transformer-models

Liu, Y., Zhang, X. F., Wegsman, D., Beauchamp, N., & Wang, L. (2022). POLITICS: Pretraining with Same-story Article Comparison for Ideology Prediction and Stance Detection. *Findings of the Association for Computational Linguistics: NAACL 2022*, 1354–1374. https://doi.org/10.18653/v1/2022.findings-naacl.101

Maladry, A., Lefever, E., Van Hee, C., & Hoste, V. (2023). A Fine Line Between Irony and Sincerity: Identifying Bias in Transformer Models for Irony Detection. *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, 315–324. https://doi.org/10.18653/v1/2023.wassa-1.28

Nadeem, M., & Raza, S. (2023). Detecting Bias in News Articles using NLP Models. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1224/reports/custom_116661041.pdf

Pennycook, G., & Rand, D. G. (2019). Fighting misinformation on social media using crowdsourced judgments of news source quality. *Proceedings of the National Academy of Sciences*, *116*(7), 2521–2526. https://doi.org/10.1073/pnas.1806781116

Pew Research Center. (2024, September). News Platform Fact Sheet. Retrieved May 24, 2025, from https://www.pewresearch.org/journalism/fact-sheet/news-platform-fact-sheet/

Rodilosso, E. (2024). Filter Bubbles and the Unfeeling: How AI for Social Media Can Foster Extremism and Polarization. *Philosophy & Technology*, *37*(2), 71. https://doi.org/10.1007/s13347-024-00758-4