Utilizing Machine Learning Techniques to Detect Bias in News Media

Colin D. Kehoe

University of Missouri-Saint Louis

CMP SCI 4390: Deep Learning

Author Note

I can be contacted at (636) 484-1744, or by email at kehoecolin012@gmail.com.

Abstract

In contemporary society, news media has become increasingly important to the lives of everyday people. In a functioning democracy, people must have access to information regarding important events happening in the world around them, especially politics. However, an increasingly common issue among the news media of the modern world is that of the proliferation of politically biased information among news media.

This study aimed to develop a machine learning model capable of discerning the political leaning of a news article based on its content. This project was conducted using data gathered by a team that conducted similar research at MIT, which demonstrated the effectiveness of BERT models in language processing.

This model takes a more traditional approach to deep learning, utilizing a simple Sequential model to achieve moderate success in a field which is still relatively new. Following preprocessing and tokenization of the text data, the model, configured with

The model displayed a propensity to favor conservative bias in its predictions, which may warrant further investigation into the content of the articles, and how that may affect the outcome of the model. Despite encountering challenges in feature selection and configuration, the project underscores the potential of machine learning in identifying and mitigating political bias in news media. Future endeavors could focus on refining model architecture and incorporating additional features to enhance performance.

As society progresses into an increasingly digitized era, the importance of combating political bias in news media grows ever more pronounced. This project stands as a proof of concept to the utility of data science and machine learning in addressing this issue, and calls for continued development in this vital domain for the future.

Utilizing Machine Learning Techniques to Detect and Combat Bias in News Media

In today's interconnected world, news media plays a pivotal role in shaping public opinion and fostering informed discourse, particularly in politics. However, the proliferation of biased information within news outlets has become an increasingly prevalent issue. The dissemination of biased reporting not only undermines the integrity of journalism but also poses a threat to the democratic process by influencing public perception.

This study is an endeavor to harness the power of machine learning techniques to detect bias in political news articles. By developing a model capable of discerning political leanings present in journalism, I aim to empower readers to critically evaluate the information in front of them and mitigate the impact of biased reporting on public discourse.

Background

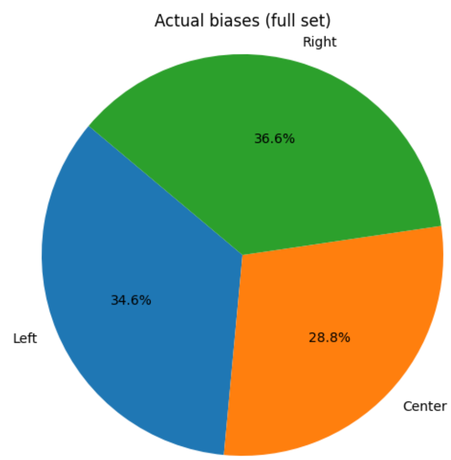
This project caught my eye as a result of how politics has become a war of propaganda. In history, people have attempted to spread their political philosophy through their professional mediums. The media is no exception. News articles and political opinion columns have always been inherently influenced by the partisan leanings of the author. On both sides of the political spectrum, people use their positions of influence and power to spread their own ideology, not outright, but by utilizing biased reporting to lure people to their own agenda. I was interested in seeing if it was possible to teach a model to identify when an article had a political agenda injected into it.

The Dataset

The dataset utilized in this project came from the website *paperswithcode*. A similar project called *We Can Detect Your Bias: Predicting the Political Ideology of News Articles* was conducted on this same dataset by 4 researchers from MIT: Ramy Baly, Giovanni Da San Martino, James Glass, and Preslav Nakov, to show that people's biases can be easily detected and exposed from their work. Their work was focused on demonstrating the power of BERT models in language processing through the medium of teaching one of them how to predict the partisan leaning of an article based on its content (among other metadata). This dataset consists of 37,554 articles, curated and manually annotated for political ideology – those being left, center, or right.

Distribution of biases

The distribution of biases within the dataset is shown in the chart to the right. One may notice that the spreads are not equal. There most represented ideology is conservative, with 13,734 articles having a right-leaning bias. Next is a liberal bias, being found in 13,005 articles. Lastly are the articles found in the center of the political spectrum, with a smaller count of 10,815 articles within the data set falling into this category.



Article sources

The 37,554 articles within this dataset came from 491 sources across the political spectrum. The 5 most prolific sources within the dataset are listed to the right. One thing you may notice is that the distribution of biases within each source is not very widespread. Most sources within the dataset are directly linked to a single bias. This made the source an excellent choice of a feature to feed the model.

Source	Left	Center	Right	Total
CNN (Web News)	2905	0	0	2905
Washington Times	0	0	2886	2886
Politico	2493	0	0	2493
Fox Online News	0	0	2047	2047
NPR Online News	0	2012	0	2012

Article Authors

The dataset consisted of 12,355 different author entries, and 9.668 of the articles did not have an author listed. Not having an author listed could cause issues with the data, however authors almost always wrote within a singular bias, without changing their stance much. This made the author a useful feature for the model, despite the existence of articles which lacked an author.

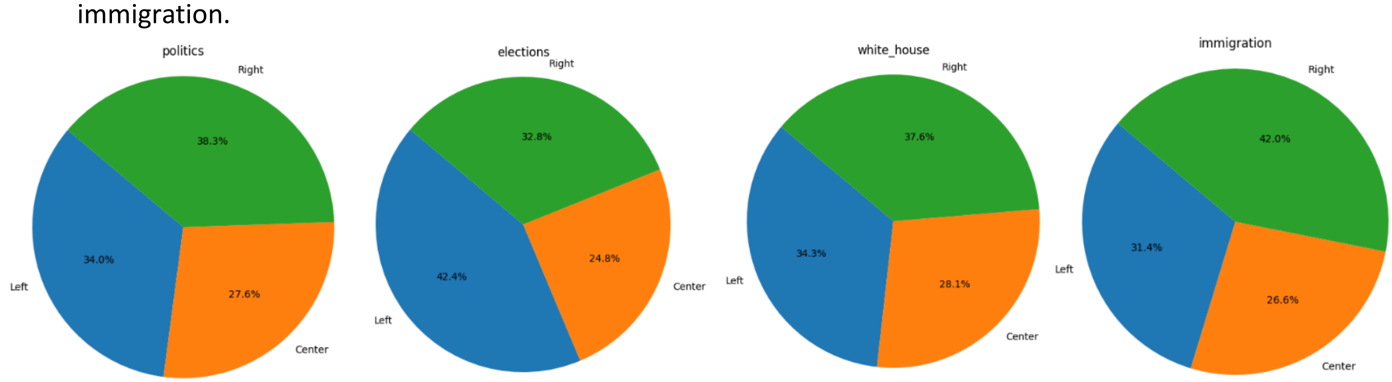
The most common author was Cortney O'Brien, who appears 922 times within the dataset, either as the sole author or co-author of the article. Her work almost always had a conservative bias. The second most common author was Matt Vespa, who appears in 907 articles, followed by Tobias Hoonhout, who appears in 855 of the articles.

Article Titles

The article title was an obvious choice of feature to feed the model. There isn't much to go over in this area from a data science angle, however the article title proved to be effective in improving the model's accuracy, which should not be a surprise. An article's title contains information that signals to a reader what is inside, and that typically includes the bias of the content.

Article Topics

Across 108 topics, the four most common are shown below, with their biases shown in the charts. Notable outliers are a left-lean on the topic of elections, as well as a right-lean on the topic of immigration.



Article Content processing

The content of each article was vectorized utilizing the CountVectorizer function provided by SciKit's Feature Extraction package. The vectorizer takes the content provided and converts it to a tensor of number values representing the number of times each word appears within each article. Common English words that may hold little meaning to what the model is designed to detect are removed, such as 'the', 'as', and 'how'.

Final Input to the Model

The final input that is fed to the model is a 2D tensor, with the row representing an article, and each column representing a word within the data, the title, the author, the topic, and the source. This is then fed to the model, and the model is asked to predict the bias of the article based on these features in the form of an integer 0, 1, or 2. '0' means left, '1' means center, and '2' means right.

Implementation

Data Source

This project was conducted on a dataset obtained from the website *paperswithcode*. The dataset consisted of 37,554 articles, each in the form of a .json file. Each .json file consisted of the following features: the topic, source, bias score, URL, title, publication date, author, content, source domain URL, and an ID.

After testing and trial and error, the final data features used consisted of the article content, author, title, topic, and source, for a total of 5 predictors, as explained in the previous section. The publication date, as well as source URL, were considered. However, as a result of a few entries within the dataset not having values in these features, if the model was asked to predict the bias of an article that had a blank publication date or source URL, it would become confused, lowering the accuracy. This same issue was encountered in the project conducted by the team at MIT.

The chosen features were preprocessed, and the text tokenized. English stop words were removed, and the article content was vectorized, before being fed to the model as a tensor of numbers representing this data. The data was split into two sets, one for training and one for testing. The training set consisted of 30,043 articles, and the test set consisted of 7,511 articles.

Model Configuration

A few different configurations were tested for this model. All of which were Keras Sequential models. The first setup consisted of three layers, the first a Dense layer with 128 neurons, and relu activation. The second layer was a dropout layer with the ratio set at 0.2. The next was a Dense layer consisting of 64 neurons, also with relu activation. The output layer predicted 1 of 3 classes (0, 1, 2) using softmax activation for multi-class classification. The optimizer used was *adam*, and the loss function used was the *Sparse Categorial Crossentropy* function. The model was trained over 30 epochs with a batch size of 32. A model checkpoint callback was used, to save and store the best weights from previous iterations, giving the model a head-start when learning.

The second-most successful configuration that was tested consisted of 5 layers, with 256, 128, and then 64 neurons, and dropout layers between the first two dense layers set at a ratio of 0.2. Similar to the previously described configuration, this model also used *adam* and *Sparse Categorical Crossentropy*, and was trained over 30 epochs with a batch size of 32, with a model checkpoint callback.

Performance Evaluation

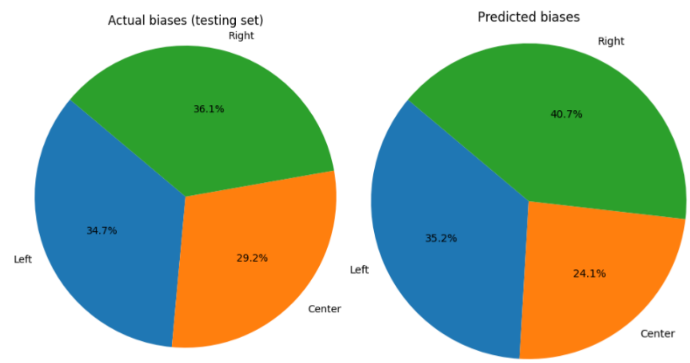
In order to evaluate the performance of this model, I utilized the reported f1-score of the model following a set of predictions being made. All results were drawn in comparison to the 0.655 f1-score that the MIT team achieved with a traditional LSTM model, and their .802 f1-score which they achieved with a BERT model. This comes with the caveat that BERT models take significantly longer to train than a simpler deep learning model, taking upwards of 20 minutes per epoch, as opposed to 2 seconds per epoch with my setup.

This score should signal that the technology to accurately detect the bias of news articles is not perfected yet. Language processing, on such a nuanced and complex task, is difficult. A baseline of accuracy is in the range of 33 to 40 percent, which can be interpreted as random guessing, depending on the training and testing split within the data.

Results

A Broad Overview

The most successful model was the first described, which consisted of a 128 neuron Dense layer, a dropout layer set at a ratio of 0.2, and a 64 neuron Dense layer, followed by an output layer with softmax activation. A snapshot of the testing dataset from the most successful run of the model is shown to the right, as well as how the model predicted based on the content provided. You may observe that the distribution is similar, however there is a lean to the right on the predicted biases, at the cost of articles that exist in the center of the spectrum. This issue will be addressed in the next section, which goes over predictions based on bias.



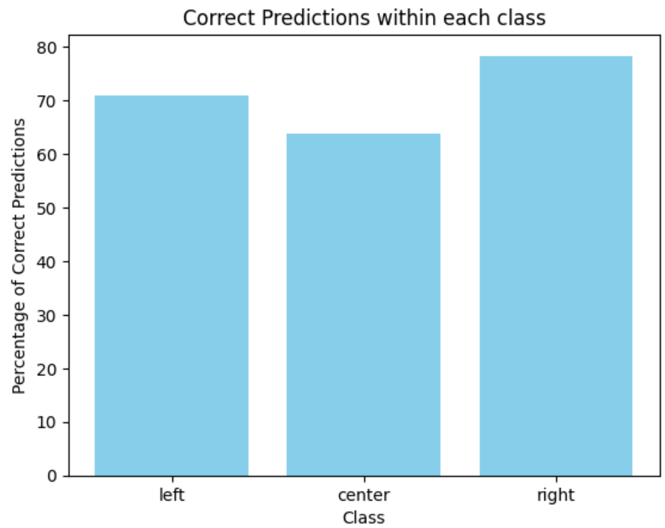
The macro-f1 score of this iteration of the model was a modest 0.71, higher than the 0.655 that the MIT team achieved with an LSTM model, but lower than the 0.802 that they achieved with a BERT model. The precision and recall statistics were fairly balanced, which indicates that the model stands at a

	precision	recall	f1-score	support
0	0.70	0.71	0.70	2603
1	0.77	0.64	0.70	2195
2	0.70	0.78	0.74	2713
accuracy			0.72	7511
macro avg	0.72	0.71	0.71	7511

	precision	recall	f1-score	support
0	0.68	0.71	0.69	2603
1	0.81	0.48	0.60	2195
2	0.62	0.80	0.70	2713
accuracy			0.67	7511
macro avg	0.70	0.66	0.66	7511
weighted avg	0.70	0.67	0.67	7511

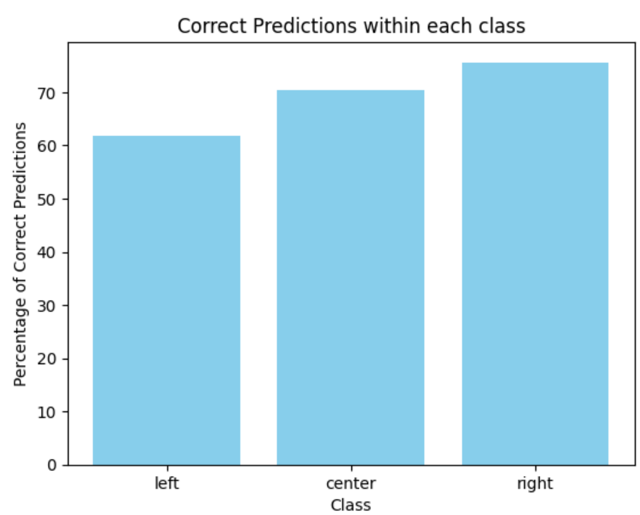
stable place in its development, which could be improved on steadily. The statistics for the second most successful iteration of the model is also shown, which achieved a notably lower macro f1-score, mostly as a result of a lower recall score.

Predictions Based on Bias



After looking at the bigger picture, we may take a dive into how the model performed on a case-by-case basis for each ideology. The model performed best on articles that had a conservative bias, and worst on articles that existed in the center of the political spectrum. The accuracy of the model in predicting a conservative bias within an article oftentimes pushed 80 percent, which represents a 10-percentage point difference over its accuracy in detecting a liberal bias, and a 17 to 20 percentage point difference over the center.

The reasoning for this is likely the slight overrepresentation of conservative articles within the dataset. It is reasonable to assume that this overrepresentation allowed the model to associate more features with a conservative bias within the articles. As a result, the model is simply more likely to guess that an article has a conservative bias, which leads to a higher score for the accuracy on this bias. Indeed, if we balance out the data, we see a change in how accurate the model is at identifying articles that exist in the center of the spectrum. See an example to the right. Keep in



mind that it was not conducted on a model with a modest amount of training and could be very much improved upon, mainly in detecting liberal bias.

Conclusion

In conclusion, this endeavor to identify political bias in news media through machine learning represents a crucial step in safeguarding the integrity of the dissemination of information to regular people. This project has made significant progress in developing a model capable of discerning the political leaning of news articles.

Despite encountering challenges in both feature selection and model configuration, the achieved f1-score of 0.71 represents a moderate success in showing that machine learning can be used as a tool to combat biased reporting in the modern world. As time progresses, model accuracy will continue to improve. Looking ahead to the future, researchers should focus on enhancing model architecture, incorporating additional features, and exploring alternative approaches to address the nuanced and complex nature of this problem. By leveraging advancements in both machine learning and data science, we can continue to advance the cause of objective reporting and ensure that individuals have access to diverse and unbiased information.

Ultimately, this project serves as a proof of concept that machine learning can be used as a tool to encourage people to confront bias head-on and uphold the fundamental principles of journalism and democratic discourse. As the MIT researchers titled their paper, we can detect your bias. Soon, it will be an everyday occurrence to check the bias of an article you are reading to know if you are trustworthy or not. Hopefully, these advancements will lead to a more unbiased world of media, news, and democracy.

References

Baly, Ramy. Da San Martino, Giovanni. Glass, James. Nakov, Preslav. (2020). *We Can Detect Your Bias: Predicting the Political Ideology of News Articles.* https://aclanthology.org/2020.emnlp-main.404