THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Media Bias Analysis

Reema Domadia

Srijon Mukhopadhyay

Professor Abdi Awl

DATS6501

6 May 2021

Table of Contents

Introduction

Background

According to Business Insider, 90% of American media is controlled by six companies: CBS, News Corps, Comcast, Disney, Viacom, and Time Warner, all of whom have white CEOs controlling the information the American people recieve. (Lutz, 2012) A prime example of media bias is during the November 2015 Paris attacks where 130 people were killed and 494 people were injured. (CNN, 2020) The following day Americans started seeing reports titled "Multiple Paris Terror Attacks Leave at Least 120 Dead". (Saliba, Emmanuelle et al., 2015) Similar attacks were carried out in Lebanon and Iraq where 255 people were injured and 60 people were killed but no mention of that was ever published. (The Guardian, 2015) Biases against such events remain often unreported in Western media.

On a domestic level, the political landscape within the United States has been drastically transformed with the emergence of digital media. Anyone can present uncontested and unvetted information directly to their audience. Due to the lack of censorship, the media has become a platform for extremists organizations to spread their ideology which can be detrimental to a democracy in some cases such as the 2021 Capital Insurrection. Political candidates have thus resorted to leveraging binary political strategies to increase political participation and voting turnout. Further, reports found that users were not interested in any news that disagreed or deviated from their accepted premises forcing traditional media outlets to line up with polarized content to keep their audience. Such instances may give birth to echo chambers that are detrimental to a democracy contingent on fair elections.

Problem Statement

Media plays a critical role in American democracy as millions of Americans rely on television to gather information about local and global happenings. People often form their opinions based on what they've seen, heard, and read in the media. While the purpose of the media is to disseminate factual information, these platforms have since pivoted to become propaganda machines or sources of polarity. Such echo chambers cause respective audiences to lose touch with other viewpoints (i.e. between conservatives and liberals) resulting in unresolvable clashes and further eroding the unity within a society. As a result, it is crucial to identify whether a certain media outlet harbors any political leaning to have greater clarity over the nature of information being received.

Problem Scope

The purpose of this project will be to identify information biases in major news sources to provide bi-partisan news. Due to the short turnaround time for this project, the media bias algorithm will be trained using the top 12 publications in a news repository created by an American dataset company, ComponentsOne, in efforts to train the model with a balanced dataset. Once the model is trained, the model will make a classification for any article so long as the words are in the model's word dictionary. Further, this project is constrained to only two classifications: Liberal and Conservative, due to the short turnaround time. This algorithm is meant to be a first prototype in tackling the incredibly complex media bias question. In the reports section of this paper, we will highlight future work that can be done to expand this project.

Literature Review

Over the years, social scientists have consistently found that an increase in polarization leads to an increase in political engagement. Gottfried et al. (2014) reported that the more polarized sections on both the Left and Right streams in the US have much greater political impact overall than sections that harbor mixed political biases. In an environment of ideological polarization, an individual's normative notions may be threatened, thus encouraging them to participate more in the political decision-making process. (Kleiner., 2019) This increase in political activity and engagement also inherently affects media habits. Pew Research Center's report on political polarization and media habits concluded that liberals tend to trust more news publications than conservatives, and both liberals and conservatives are prone to be friends with people who share their same political views. (Gottfried et al., 2014) This naturally brings to mind the formation of an echo chamber. It is thus worth questioning how much of a role media plays in encouraging — if not forming — these echo chambers.

Evidence of bias amongst media sources has been extensively documented. This bias has been found on both the Left and Right sides, depending on the media source in question. (Groseclose et Milyo, 2003) Larcinese et al. (2011) analyzed large samples of US newspapers in the 1996-2005 period and uncovered pro Democratic biases in the way unemployment was covered by media outlets when the incumbent was a Republic President, in addition to evidence of partisan biases in the way the budget deficit was covered. This was corroborated by Merkley's (2018) study which also concluded that bias in media especially exists during worsening economic conditions. The existence of this media bias is also not lost on the general public and nearly two thirds of respondents in a poll admitted to feeling that media outlets are 'politically charged'. (Groeling, 2013) However it can be argued that this awareness doesn't hamper echo chambers — in fact, it results in greater distrust of news outlets that provide different views than that of the consumer.

Modeling and detecting bias in media articles is thus a task that has gained interest and become a reality especially with the advent of deep learning. Iyyer et al. (2014) improved the then standard 'bag of words' classifier approach in favor of a Recurrent Neural Network (RNN) model that classified ideological biases at the sentence level as opposed to earlier trends of classifying at the document level. The classification at the sentence level helped utilize linguistic context that is often lost when the corpus is modeled at the document level. Dadu et al. (2020) fitted Bidirectional LSTM models with word embeddings and extracted subjective bias from the Wiki Neutrality Corpus (a corpus containing Wikipedia edits that were made to remove subjective bias). The trend of using Recursive Neural Networks and Long Short Term Memory (a specialized version of the RNN) models for classifying bias within text is not a coincidence; researchers have found success with RNN and LSTM primarily due to their ability to capture sequence data. In a text processing context, text is mostly seen as a sequence of tokens, thus making RNNs a natural choice for modeling text.

In a similar vein, Misra et Basak (2017) used unidirectional LSTM models to detect political bias within the Ideological Books Corpus (IBC) with an F1 score of approximately 0.71 (which is considered to be a good score in bias classification). In a related direction, Wang et al. (2018) also used LSTM for short text sentiment classification obtained F1 scores of over 0.8. LSTM models are thus considered to be the current state of the art in the realm of political bias classification with an F1 score of 0.7 to emulate or better.

Methodology

Dataset

The dataset used for this project was created by Andrew Thompson on April 3rd, 2020 with ComponentsOne. ComponentsOne is a publication and research group that assembles, investigates, and editorializes large datasets, with members coming from a variety of backgrounds and are located throughout the United States. The original dataset was 143,000 articles from 15 different publications but it had more articles from publications that are considered liberal than conservative. We narrowed it down to 96.7 thousand articles that range from 2014 to 2017. This gave us 48586 conservative data points and 48163 liberal data points which ensured a better balance in terms of both labels.

The dataset consisted of general news articles that were collected from different publications. We decided not to go with the more typical approach of choosing politically inclined data sets like Convote (a data set of US Congressional Speeches) and the Ideological Books Corpus (another data set containing 4026 politically charged sentences with their corresponding 'conservative' and 'liberal' labels) for this present work. Since headlines of recent articles tend to subtly carry biases within them, training our model using more politically charged sentences might affect the model's power to generalize and capture the nuances of the biases carried by the everyday headlines of news articles. Furthermore, the Convote dataset was labeled according to political affiliation in place of political bias (Democratic or Republican) which created further complications in our case since political affiliations do not always necessarily signal political bias. For example, it is very possible for a Republican to identify with liberal viewpoints and a Democrat to identify with conservative viewpoints, though of course, examples of staunchly conservative Republicans and liberal Democrats can also be pointed out.

An area of concern for us was ascertaining a way to annotate our data while maintaining neutrality on our part as analysts. Though we preferred to crowdsource, we did not have the time or resources to use public crowdsourcing methods for annotations. We were also hesitant to introduce our own crowdsourcing groups as it would risk a bias in the way the crowd was chosen. We used the website allsides.com to assign labels of 'conservative' and 'liberal' to the articles based on the media sources that published them. Allsides is a website that accepts consistent crowdsourcing information about different publications and their biases and then using its own robust statistical methodologies, assigns bias ratings to the publications. For the time and resources available to us, we decided that this would be the best immediate solution to annotate our data and prepare it for modeling.

Data Preprocessing and Feature Engineering

In our annotated dataset, we were most concerned with the headlines that we were to fit our model with and their corresponding labels (whether they were 'conservative' or 'liberal'). Standard preprocessing steps were taken to remove stop words from the data and convert words to lowercase. The words were also stemmed to ensure a uniform format of words and also ensure accurate word counts.

A train test split function from the sklearn package in Python was used to randomly split the data set into training and test data sets.The training data set retained 80% of the original data set and the test set, 20%. From both the training and test data sets, the headlines were extracted and converted to arrays of words. Using the keras package, these arrays were fit on the keras tokenizer which tokenized them and converted them into integers based on their position in the resulting word index. These integers were then converted into sequences and padded with zeros so that all the resulting sequences were of uniform shape.

The labels were also extracted from the training and test data sets. Using the Label encoder function, they were encoded to 0s and 1s (corresponding to conservative and liberal labels).

Fitting Model

For our model to better fit the data and make predictions that weren't random, it was necessary to fit it with word embeddings. Word embeddings are representations of texts that are trained on text corpula and are able to capture lexical semantics of text. (Bakarov, 2018) They can be considered to be spatial representations of text. For our purpose we used the popular word2vec model. We used pre-trained word embeddings that were fitted on the word2vec model. The pre-trained word embeddings were obtained by training the FastText model on a corpus of one million WikiNews articles. FastText is an open source classifier model that can generate text classification tasks as well as vector representations of words. (Joulin et al, 2016)

For the task at hand, an LSTM model was chosen over the RNN model. Though the LSTM model is essentially a specialized form of the RNN network, it has certain advantages over the RNN network. The LSTM network has 'memory cells' and feedback cells that allow it to retain information for longer periods of time. RNNs also suffer from the vanishing gradients problems which is a state where the gradients updates become almost negligible thus leading to insignificant parameter updates and hampering learning. The LSTM model with its specific gradient structure and frequent gate updates, is able to control for this problem. This is a key advantage that an LSTM model holds over the RNN.

The LSTM model also has the additional feature of adding a bidirectional layer. This is a layer that allows the LSTM model to access past inputs while processing future inputs in the reverse direction. This is especially useful for text sequences, as text processing in the mind is almost never linear. The ability to add a bidirectional layer in the LSTM makes it very well adjusted for modeling text and performing text classification tasks.

A basic representation of how a Bidirectional LSTM may handle text is shown below:

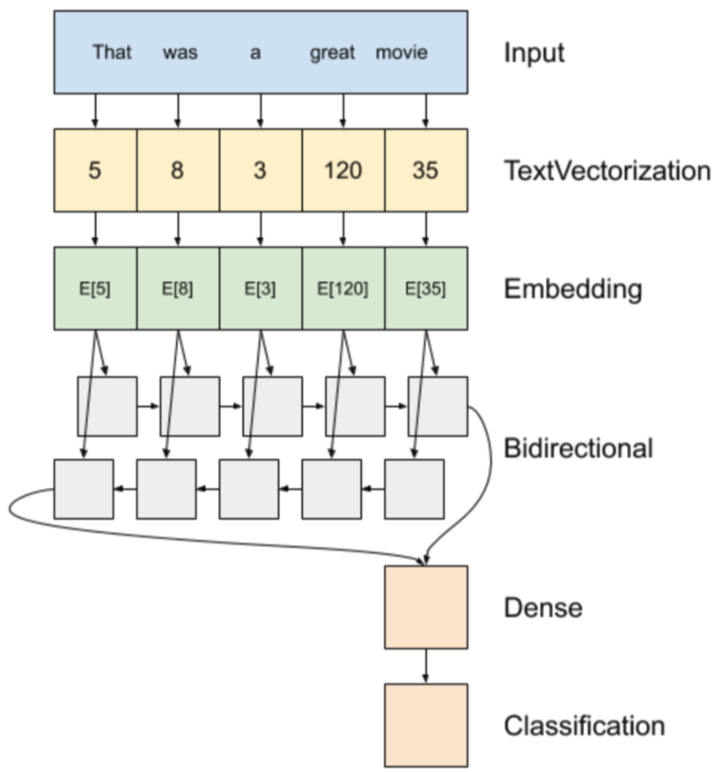


Figure 1: *Text Classification with an LSTM* (**Source:** TensorFlow)

We incorporated an LSTM model with an Embedding layer consisting of our pre trained word embeddings. A dropout layer with 0.6 retention was added to control for overfitting, followed by a Bidirectional LSTM layer with 128 neurons. The output layer consisted of a softmax activation function to generate two probabilities of the headline in question being either liberal or conservative biased.

Layer (Type)	**Shape**
Embedding Layer	24 x 300
Dropout Layer (0.4)	24 x 300
Bidirectional	24 x 256
LSTM Layer	128
Dense Layer	2

Table 1: *Model Shape*

This model was then fitted with the training data. 33% of this training data was used as a validation set for the model to update its weights. Different combinations of batch sizes and learning rates were tried out to find the optimal combination.

After trying different combinations of batch sizes and learning rates, the model provided its best fit at a low batch size of 32 and learning rate of 1e-03. In tune with recent research on optimizing batch size and learning rate, learning rate decaying was not attempted. (Smith et al, 2018) It was kept in mind that larger batch sizes might lead to lower accuracy on the test set but this may be prevented by increasing the lower rate. However, the model performed best with a lower batch size and learning rate. The better performance on the part of lower batch sizes can be attributed to the ‘generalization gap’, a phenomenon where larger batch sizes lead to degradation of the gradient updates and reduced predicting power on unseen data. (Hoffer et al, 2018)

Since the learning rate was not decayed, we chose Adam as our optimizer over the Stochastic Gradient optimizer.

Results, Analysis, & Visualization

Once the model was fitted with the training data, we evaluated it by running it on the test data set and predicting biases. The model showed an accuracy of 0.76 on the test set.
Binary classification models are generally evaluated by using F1, precision and recall score metrics and the AUC-ROC curve. (Misra et Basak, 2017)

Precision implies the proportion of the predicted instances that are actually relevant, while recall values are the proportion of relevant instances that the model is able to predict correctly. Since precision cannot be improved without decreasing recall, it is important to have another metric that provides a measure of how the model does in reducing both errors. The F1 score is the harmonic mean of the precision and recall scores of the model and is used in this case. An F1 score higher than 0.5 would imply that the model is able to control for a significant number of false negatives and false positives and can thus make reliable predictions.

On the test set, the number of false positives and false negatives were documented and precision, recall and F1 scores were generated.

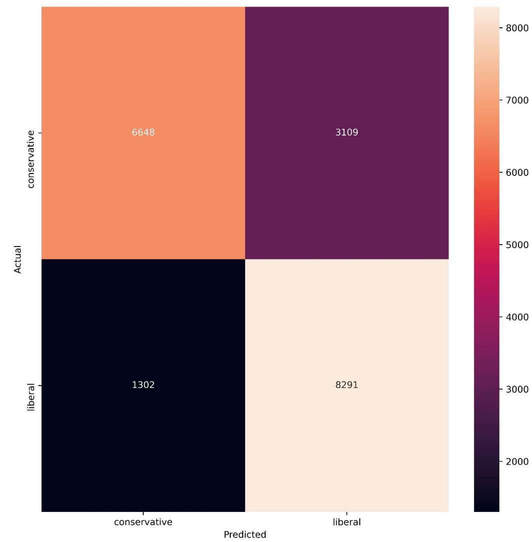


Figure 2: *Confusion matrix*

Conservative biases were predicted correctly 6648 times out of a total 9757 instances, and liberal biases were predicted 8291 times out of a total 9593 instances.

Label	**Precision**	**Recall**	**F1 Score**
Conservative	0.68	0.84	0.75
Liberal	0.86	0.73	0.79

Table 2: *Evaluation scores for LSTM model*

The F1 scores are higher than the metric of 0.7 that we considered to be our baseline. There is however, some overfitting in favor of liberal biases. However this cannot be controlled without adding more training samples. Increasing the dropout layer to any value more than the current magnitude of 0.4 would affect the model's accuracy. Overall, the model performs well and is able to distinguish between the two classes. This is further confirmed by looking at the AUC-ROC curve:

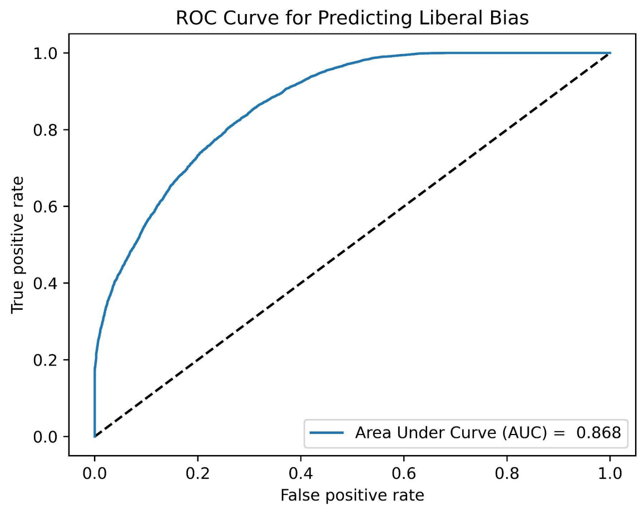


Figure 3: *AUC-ROC curve*

Higher AUC scores measure the capability of the model to distinguish between two classes. Values may range between 0 and 1, with 0.5 suggesting that the model is unable to distinguish between the classes, 1 suggesting that the model perfectly separates the two classes and 0 suggesting that the model inverts the classes (in our case it would imply that the model predicts conservative biases to be liberal and vice versa). Our score of 0.86 is thus desirable.

Visualization

This work is composed of two components: the Bias Algorithm and the Tableau Dashboard. While the bias algorithm will work to identify the degree of bias and then label its leanings, the Tableau Dashboard helps users identify key insights, trends, and patterns in bias data by article and publication. Tableau's strength lies in that .tbx files are shells for the data and can be updated by simply updating the data source and refreshing. The ability to toggle through filters (versus static pictures) enables the dashboard users to achieve greater flexibility in the data being used (so long as the variables remain the same). Furthermore, this allows users to independently extract insights that pertain to questions they are seeking to answer, compared to static visuals where users have to leverage only what is given to them. This, coupled with domain knowledge, can better equip decision-makers, researchers, and citizens with increased awareness to help them stay factually informed.

The goal of the Tableau dashboard is to identify high-level insights (Fig 4) and provide an article-level analysis (Fig 5). As it stands, the Media Bias Tableau Workbook contains three individual dashboards: Publication Bias Breakdown, Net Publication Bias, and Article Repository.

- Publication Bias Breakdown: Cumulative breakdown of publication leanings (Fig 4)
- Article Repository: Mine database for basic exploratory data analysis (Fig 5)
- Net Publication Bias: Graph the net degree of bias (Fig 6)

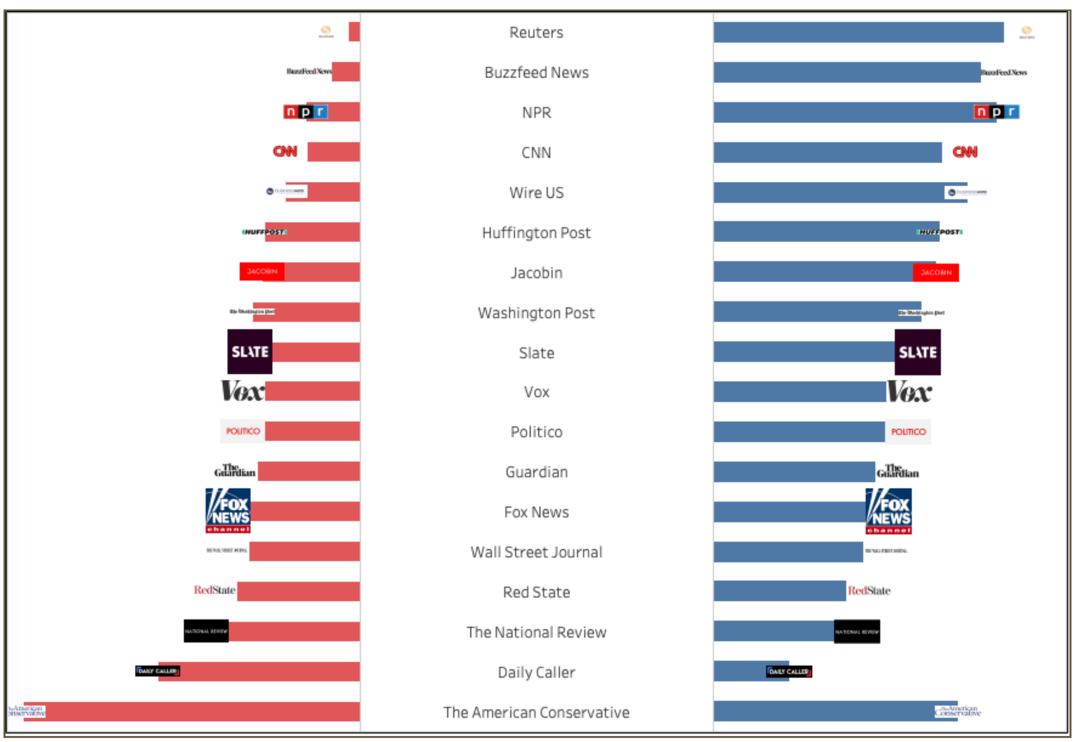


Figure 4: *Publication Bias Breakdown*

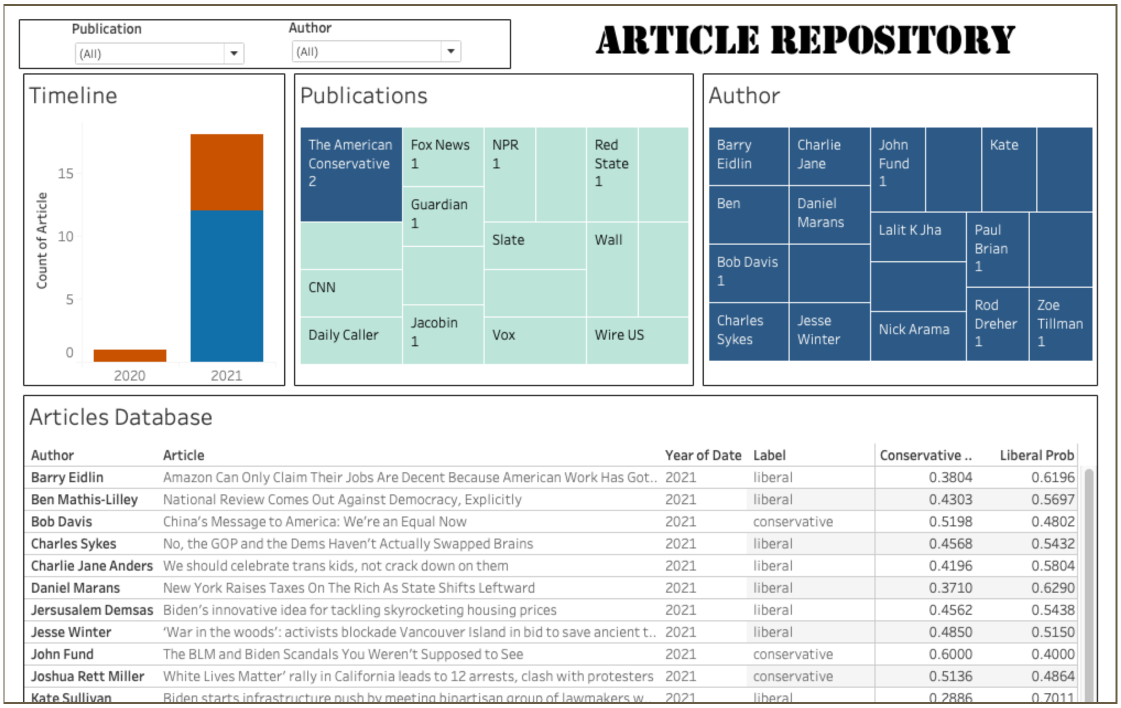


Figure 5: *Article Repository*

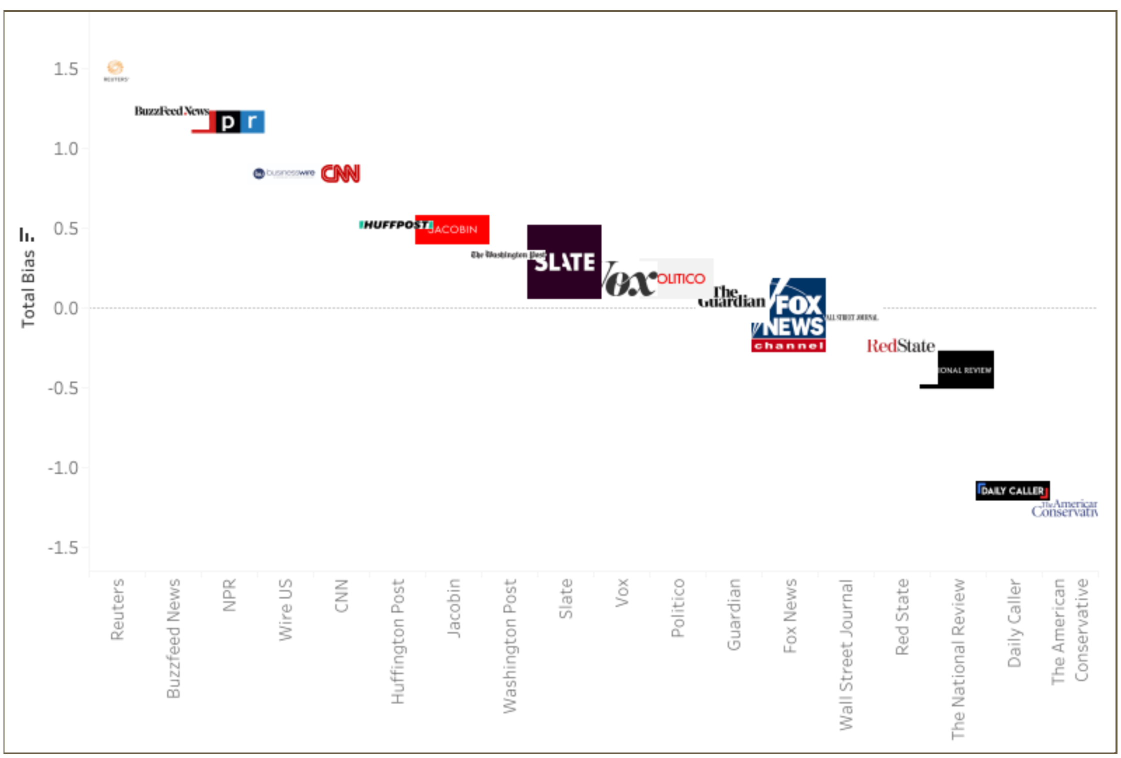


Figure 6: *Net Publication Bias Dashboard*

Conclusion

Limitations

Due to the quick turnaround time, this work was limited in scope. The goal of this project was to initiate a prototype of deciphering the bias in the media for future researchers to expand capability. As it stands, the model is only trained to classify an article or publication as “liberal” or “conservative”, however in the real world that's hardly ever the case. Further, the media bias models were only trained using the 12 publications found in the ComponentsOne Dataset which likely caused overfitting when training the Word2Vec model as it leverages polarity to determine its position on vector space. By training using a larger and more diverse dataset, the model will be able to control for this overfitting and make better predictions.

Another limitation was the limited control over the degree of politicization of the headlines we used to train our model. Though we stand by our decision to use more general articles rather than outrightly political speeches and transcripts, we were also confined to the articles that were present in our dataset without any flexibility over the type of headlines that were being fit on the model. We also considered creating our own general dataset but there were concerns over time and potential risk for bias on our part. This would be worth exploring in future work.

Lastly, the other test dataset visualized in tableau were random articles hand-picked at the time of testing. The level and degree of politicization here too, was a limitation as they were entirely dependent on a combination of inevitable human bias and Google’s search algorithm.

Future Research

More Data: Publications and Articles

This project can be expanded to use more data (publications and articles) to train the model to ultimately be able to add additional leaning labels such as “indeterminate” or “moderate”, which will likely allow the model to stop categorical overfitting.

Increased Language Capability

Language-intelligence to enable the model to be able to detect biases in foreign language media. This helps politicians and decision makers gauge perceptions of key U.S. foreign policy and decisions, especially as the world increases interconnections — every decision has a ricocheting effect.

User Interface

Due to the nature of this project, a potential avenue of expansion could be a web-based biased calculator using the bias algorithm. The goal would be to expand the audience so that users could drop a link into the input and the calculator would determine how biased the article is. This would be an opportunity for the website and the algorithm to continuously learn and improve itself. Furthermore, due to Tableau’s embedding-friendly capability, the dashboard can be leveraged for users to visualize their results and extract other insights from the database that the website would create by saving data that users input.

More Contemporary Word Embedding Models

We used very traditional and well tested word embedding models for our purpose. Newer deep contextualized word embeddings have been developed that are able to account for a stronger amount of linguistic variation and semantics. (Peters et al, 2018) Future work would do well to incorporate these new developments in our model.

References

1. CNN Editorial Research. (2020, November 4). *2015 Paris Terror Attacks*. CNN. https://edition.cnn.com/2015/12/08/europe/2015-paris-terror-attacks-fast-facts
2. Shaheen, K. (2018, April 14). *Isis claims responsibility as suicide bombers kill dozens in Beirut*. The Guardian. https://www.theguardian.com/world/2015/nov/12/beirut-bombings-kill-at-least-20-lebanon
3. Lutz, A. (2017, May 12). *These 6 Corporations Control 90% Of The Media In America*. Business Insider. https://www.businessinsider.com/these-6-corporations-control-90-of-the-media-in-america-2012-6?international=true&r=US&IR=T
4. Saliba, E., Euronews, The Associated Press, Ing, N., & Fieldstadt, E. (2015, November 14). *Multiple Paris Terror Attacks Leave at Least 120 Dead*. NBC News. https://www.nbcnews.com/storyline/paris-terror-attacks/french-police-report-paris-shootout-explosion-n463186
5. Mitchell, A., Gottfried J., Kiley J., & Matsa K.E. (2014). Political Polarization and Media Habits. Pew Research Center. https://www.journalism.org/2014/10/21/political-polarization-media-habits
6. Kleiner, T. (2019). Does Ideological Polarization Mobilize Citizens?. *European Political Science, 19,*573 - 602.

   https://doi.org/10.1057/s41304-019-00228-y

7. Groseclose, T., Milyo, J. (2003). A Measure of Media Bias. *The Quarterly Journal of Economics, 120(4),* 1191 -1237.

   http://www.jstor.org/stable/25098770

8. Larcinese, V., Puglisi, R., & Snyder, J. (2011). Partisan Bias in Economic News: Evidence on the Agenda Setting Behavior of the United States. *Journal of Public Economics, 95(9),* 1178-1189.

   10.1016/j.jpubeco.2011.04.006

9. Merkley, E. (2018). Partisan Bias in Economic News Content: New Evidence. *American Politics Research 47(6)*

   https://doi.org/10.1177/1532673X18821954

10. Groeling, T. (2013). Media Bias by the Numbers: Challenges and Opportunities in the Empirical Study of Partisan News. *Annual Review of Political Science, 16,*129 - 151.

    https://doi.org/10.1146/annurev-polisci-040811-115123

11. Iyyer, M., Enns, P., Boyd-Graber, J., & Resnik, P. (2014). Political Ideology Detection Using Recursive Neural Networks. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, *1: Long Papers*, 1113–1122. https://doi.org/10.3115/v1/p14-1105

12. Dadu, T, Pant, K, & Mamidi, R. (2020). Towards Detection of Subjective Bias using Contextualized Word Embeddings. *Companion Proceedings of the Web Conference 2020*, 75–76. https://doi.org/10.1145/3366424.3382704

13. Misra, A., Basak,S. (2017). Political Bias Analysis. *CS224n,*8.

14. Wang, J., Liu, T., Luo, X & Wang, L. (2018). An LSTM Approach to Short Text Sentiment Classification with Word Embeddings. *The 2018 Conference on Computational Linguistics and Speech Processing,* 224-223.

15. Bakarov, A. (2018). A Survey of Word Embeddings Evaluation Methods. arXiv:1801.09536v1

16. Joulin, A., Grave, E., Bojanowski, P., & Mikolov T. (2016). Bag of Tricks for Efficient Text Classification. arXiv:1607.01759v3

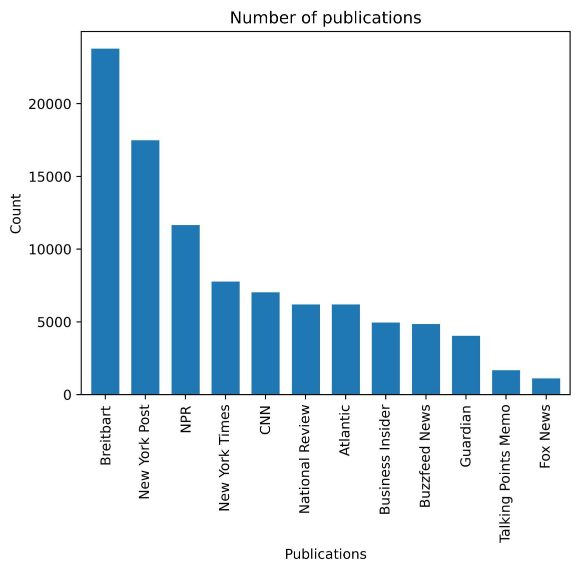
17. *Text classification with an RNN: TensorFlow Core*. (n.d.). TensorFlow. https://www.tensorflow.org/tutorials/text/text_classification_rnn

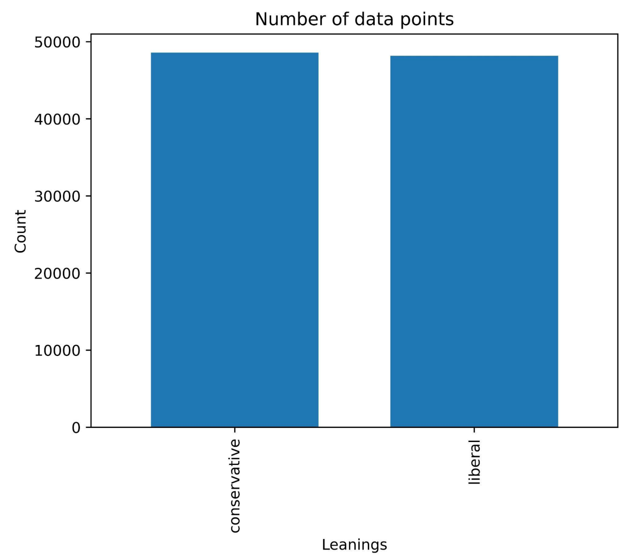
18. Smith, S., Kindermans, P., Ying C., & Le, Q.V. (2018). Don't Decay the Learning Rate, Increase the Batch Size. *ICLR 2018. arXiv:1711.00489v2*

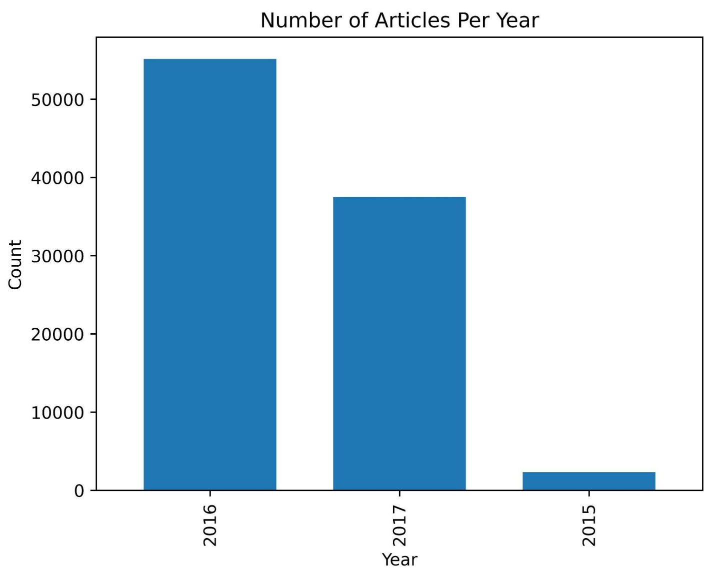
19. Hoffer, E., Hubara, I., & Soudry, D. (2017). Train Longer, Generalize Better: Closing the Generalization Gap in Large Batch Training of Neural Networks. *Advances in Neural Information Processing Systems 30,* 1729 - 1739.

20. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *1 (Long Papers)*, 2227–2237. https://doi.org/10.18653/v1/n18-1202

Appendix






Number of Articles Per Year
50000
40000
30000
20000
10000
0
Count
2016
2017
2015
Year