SAARLAND UNIVERSITY

Controllable NLG Software Project: Removing Political Bias from News Headlines

Alice Chase
Matriculation Number: 7007281
March 29, 2022

1 OVERVIEW

This project is mainly based off of the paper Automatically Neutralizing Subjective Bias in Text[1]. My project aims to test the bias neutralizing algorithm on an out of domain dataset I crafted and analyze the results via a classifier.

Besides being technically relevant to the course, this technology also has important social implications due to the prevalence of filter bubbles and fake news online. One potential way to fight against the disinformation epidemic is to have ways to automatically remove bias from sources.

2 DATASET

I gathered my data from Allsides[2]. This website aggregates news sources across the political spectrum and provides rating for where the article falls on it (from left to right) through a variety of methods such as editorial reviews and blind surveys. The political leaning categories an article can be classified in are Left, Right, Center, Lean Left, and Lean Right.

I made a balanced set of headlines belonging to center, left, and right sources, with 2939 in each category or 8817 total. If I exclude 'Lean X' headlines (center always stays the same), then headlines belonging to 'Left' reduces to 1031 and 'Right' to 1678. From these Left and Right headlines, I took 250 each. Using the id column, I then removed all of those headlines

(including the ones for center) if it matched the id in either the left or right subset. These are the 500 headlines I ran through the MODULAR algorithm and then tested on with the classifier. The rest of the headlines were used for training and validating the classifier.

3 METHODS

Three repositories by previous users on Github were combined for the final project. I modified code from each to fit my need. I also wrote three different preprocessing scripts to get the data into the forms I needed, which can be viewed on my repository.

3.1 WEBSCRAPING

I modified csinva[5] webscraping code by simplifying the output and amount of information downloaded as it was originally used to generate an app to easily display data. I only pulled the headlines and reformatted it into csv documents with the following columns: left_story_title, left_story_leaning, left_story_topic, right_story_title, right_story_leaning, right_story_topic, center_story_title, center_story_leaning, center_story_topic.

Titles are the headlines, topic is the category (such as energy, LGBTQ, etc.) and Leaning is stated above in section 2. An example of headlines for the same topic but from different outlets is given below:

Center:	Biden Tells States to Make All Adults Eligible for Covid-19 Vaccine by May 1
Lean Right:	Coronavirus,Biden delivers prime-time address on anniversary of COVID-19 pandemic
Lean Left:	Biden directs states to make all adult Americans eligible for vaccine by May 1

Center:	"Idaho governor signs GOP's anti-transgender bills, setting up likely legal challenges"
Right:	"Idaho Gov. Signs Law Barring Biological Males, 'Transgenders,' From Female Sports"
Left:	Idaho Governor Signs 2 Bills into Law Denying Trans People Basic Rights

3.2 BIAS REMOVAL

The second part involved using the github repository[3] by the authors of the bias neutralization paper[1]. The authors created two models for removing bias which they labelled as MODULAR and CONCURRENT. As I didn't have the computational resources to train a model from scratch, I used the pretrained model which they only released for MODULAR. This model architecture consists of a BERT-based detection and LSTM-based editing. The authors also stated that it is better at reducing bias than the CONCURRENT algorithm, although the output tends to be less fluent.

After running some mock tests, I found that the Modular model is not that sensitive, in that most headlines in 'Lean left' or 'Lean right' were not changed. Also from the example

above, it is hard to discriminate even from the human perspective of whether or not there is bias in some of the 'Lean X' headlines. Therefore, I decided to only pass in headlines which were labelled as 'Left' or 'Right' to the model. The model also outputs an overall BLEU score. Originally this was used to compare it to a Gold Standard sequence and in my case there is none to compare it to so it was against the original headline, which gives us an idea how much the model is changing the sentences. The BLEU scores were high, 87 for both left and right sentences and many sentences did remain unchanged. Here are some example outputs:
Change (input is a Left headline):

Input:	Don't blame fate for Beirut's cruel tragedy
Output:	Don't blame fate for Beirut's tragedy

No Change (input is a Right headline):

Input:	Senate overwhelmingly passes criminal justice overhaul
Output:	Senate overwhelmingly passes criminal justice overhaul

3.3 CLASSIFIER

The classifier used was a pretrained simple 4-layer Convolutional neural network[4]. While there were a couple classifiers of various architectures that contained three classes (left, right, center), binary classifiers (bias, not-biased) tend to have more accuracy so I decided to do binary classification since bias removal and not political party affiliation detection was the ultimate goal.

The dataset for the classifier[4] was initially trained on a private dataset of news articles on which the authors obtained 96% accuracy for their test set. Because their dataset used full articles, I made a baseline for myself by splitting the original headlines I scraped (no bias removal algorithm applied to anything) into train, validation, and test sets. This resulted in 90% for the test set. I removed the 'Lean X' headlines for this.

I reset the model weights back to the ones provided by the author[4] and ran a couple more experiments, this time testing on a set containing only headlines that had been run through the MODULAR algorithm. For the training and validation sets, I tried methods where I both included 'Lean X' headlines and excluded them in the training and validation sets (where they were labelled as having bias). I also tried training the word embeddings further but it always decreased accuracy. Unfortunately the results were always poor on the test set of MODULAR headlines.

Table 3.1: **Training on data that includes 'Lean X' headlines:**

Originally Right Headlines:	**27.6**
Originally Left Headlines:	**34.4**
Average Accuracy:	**31**

Table 3.2: **Training on data that excludes 'Lean X' headlines:**

Originally Right Headlines:	**34.80**
Originally Left Headlines:	**55.6**
Average Accuracy:	**45.2**

4 Discussion and Results

The poor results from the classifier aren't surprising given that a lot of the headlines were unchanged from the MODULAR algorithm, however it is interesting that headlines that had originally been right-aligned always did significantly worse than left ones, even though for the bias-removal algorithm the BLEU scores were approximately the same for both sets. It would be interesting to do a few different 'baseline tests' for headlines if I had also separated the test set for left, right, and center (I had it all together) to see if the accuracy differed between groups on human-written data and then I could have a clearer picture if the problem lies with the bias-removal algorithm or the classifier when it comes to the accuracy discrepancy between left and right headlines with bias-removal. Another possibility to increase classifier accuracy is including 'Lean X' headlines in the training set but labelling them as belonging to the center group.

Another issue I encountered is that the bias-removal algorithm and classifier both did different pre-processing. The MODULAR algorithm did heavy pre-processing while the classifier merely split over sentences. To get the data into shape for the classifier I had to write a script to reformat the MODULAR output but it likely degraded the output further.

References

[1] Reid Pryzant et al. "Automatically Neutralizing Subjective Bias in Text". In: *AAAI*. 2020.

[2] *AllSides*. URL: `https://www.allsides.com/`.

[3] *Neutralizing Bias*. URL: `https://github.com/rpryzant/neutralizing-bias`.

[4] *News Bias*. URL: `https://github.com/natel9178/CS230-news-bias`.

[5] *News Title Bias*. URL: `https://github.com/csinva/news-title-bias`.