Idean Labib ilabib1@jhu.edu (601.466)
Eddie Heredia <eheredi1@jhu.edu> (601.466)

This project is a Political Bias Comparer. Its primary focus is to detect the level and direction of political bias in a news source, or any other article.

Full Corpus tar file: https://livejohnshopkins-my.sharepoint.com/:u:/g/personal/ilabib1_jh_edu/EfAgsTonBlxCvDPKFNjZl1IB7yV0InOecH77LuAYYTVTLQ?e=dGLQMd

The code can be run as a perl script with perl main.prl. This assumes data has already been preprocessed using init_model.pl. This will output the biases of the documents in the corpus_list.

We are proud of this project for several reasons:

- We felt that this topic would be very relevant to modern times. With so much talks of fake or biased news sources, there are very little to no credible, neutral sources that judge bias in journals. It seems that nobody can agree on the bias of Fox News and CNN. It was this opportunity that led to our idea.
- We feel that we mixed many topics of IR and Web Agents in a fashion that complements the course well, while using our creativity along the way to determine how to approach challenges. The usage of the traditional bag-of-words model, SVD, and a link crawler are the highlights of this project that couldn't have been done without the course topics.
- A strength of our project is the preprocessing, and we feel that this should not be overlooked. For the processing of data, we had to implement a link crawler that would crawl non-self-referencing links within a domain. This allowed us to download our 9 corpuses. Following this, we had to convert the html files to the relevant corpuses while stemming and tokenizing (involved fixing a bug in the nstemmer.c starter code for HW2). After these steps, we can finally load the corpuses into a bag-of-words model with 'init_model.prl', only to output the documents' average term weight vector to data files. The purpose of this is to then load these caches on demand in the main perl script to then perform SVD on.
- The project is interactive. We allow users to specify their document URL, and then we compare it to the known biases from preprocessed corpuses.
- Completely ethical. We are using a Perl LWP::RobotUA object which means we obey the robots.txt of the websites we crawl. If a journal doesn't want us crawling them, we won't. We also ensure the delay between requests is reasonable, so we don't incapacitate any servers of potentially smaller journals such as ThinkProgress.
- We used a range of journals for our original corpuses. These journals vary differently in how they are perceived. Some have obvious conservative bias (Brietbart), whereas some have blatant liberal bias (ThinkProgress or DailyKOS). Some are considered mostly neutral and factual and not as analytical (Bloomberg), while others are analytical and have widespread disagreement in bias (CNN or Fox News). Using a range of journals allows us to express a large variance of political bias assignments.

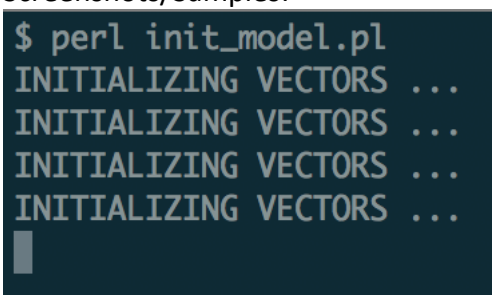
Limitations to this project are:

- We realize we are introducing some bias into the project by using Brietbart as a seed on the right, and Thinkprogress as a seed on the left. However, Breitbart is commonly regarded as very conservatively biased. Resolving this could be as simple as collecting words from articles and quotes of GOP leaders that formally affiliate themselves with a party.
- There may be lots of noise in our data. This could explain the strange results we found. This could be because of of useless "terms" picked up by the html2corpus.pl script we

used. My theory is that some journals use similar HTML platforms and SVD is picking that correlation up. Some articles have more useless terms than others (CNN). This could be resolved with more time by revisiting the html2corpus script and improving it case-by-case for each journal to extract only the representative body content.

- It does seem like SVD is picking up the wrong latent trends in the data. It may be picking up how extreme documents are rather than how biased they are. We think that an improvement could be to cluster by topic first before performing SVD on each cluster, and averaging the similarity results. This way the articles that are similar will then be compared.
- We were planning on allowing only a single URL to be provided by the user, who then may get a poor result depending on the URL provided. However, the SVD results did not turn out reliable enough to include this. If URLs could be submitted, it would have poor results because for example, if the user provides a short article that is only a few sentences, it may be very hard to judge its similarity with the other entire corpuses after SVD. Longer articles may have the opposite, and a positive effect. This could obviously be improved by crawling the URL provided and processing it the same way we preprocessed our 8 starter corpuses. This would be a good extension to the project.
- A web interface would be a nice extension. Currently, the input is provided to standard input in a perl script that is running the main program. Integrating this with a web interface would be another extension given more time.

Screenshots/Samples:



```
$ ls
Makefile        files           make_hist.prl   stemmer         token1.c        token1.o
build.sh        html2corpus.prl robot_base.pl   token1          token1.l        tokenize
$ perl html2corpus.prl
Syntax:
  ./html2corpus.prl ./directory outputfile
$ perl html2corpus.prl files huffington_post
Looking at files...
Found 3955 files in files/ ...
Reading: ..
Reading: 11666.html
Reading: 12949.html
Reading: 20838.html
Reading: 19606.html
Reading: 9315.html
Reading: 2109.html
Reading: 17618.html
Reading: 5336.html
Reading: 8011.html
Reading: 5273.html
Reading: 22940.html
Reading: 10831.html
Reading: 21395.html
Reading: 8912.html
Reading: 21815.html
Reading: 2365.html
Reading: 17024.html
Reading: 2735.html
Reading: 14258.html
Reading: 23081.html
Reading: 1119.html
Reading: 13572.html
Reading: 22385.html
Reading: 13821.html
Reading: 2220.html
Reading: 7022.html
Reading: 20642.html
Reading: 20212.html
Reading: 9895.html
Reading: 17161.html
Reading: 15672.html
Reading: 22151.html
Reading: 5731.html
Reading: 4823.html
Reading: 4570.html
Reading: 1288.html
```

```
$ perl main.prl
BIASES:
breitbart: 1
thinkprogress: -1
daily_kos: -0.00839766640300443
cnn: -0.158938191943507
fox: 0.0413087473999217
washington_post: 0.112266689248541
huffington_post: 0.027987143795755
bloomberg: 0.0162619366343068
```

As mentioned earlier, the results were not what we expected. We expected clear trends in the bias of journals, but most were centered around 0 (-1 being left, +1 being right). Although CNN was founds slightly left and Fox was found slightly right, Washington Post was slightly right of 0 which doesn't seem accurate. We think this may be because of significant noise in the data, or maybe we had to cluster by article topic first. We will most likely revisit this project with more time to try and squeeze more representative results out. The corpuses and corpus processing are very useful to keep.

Credits:
Almost all of the scripts in the newcorpus/ directory were originally provided by Professor Yarowsky for past homework assignments. However, many did not work on my architecture and also had other bugs such as a letter immediately after a period would not be stemmed

properly. These bugs were fixed and the whole script was modified for our use case. Additionally, the starter code for vector1.prl was provided by Professor Yarowsky for Homework 2. However, vector1.prl has changed significantly from its original form.