Idean Labib [ilabib1@jhu.edu](mailto:ilabib1@jhu.edu) (601.466)

Eddie Heredia <[eheredi1@jhu.edu](mailto:Eheredi1@jhu.edu)> (601.?66)

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. TBD

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. 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 would be because of the amount of useless “terms” picked up by the html2corpus.pl script we used. 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.
* Only a single URL can be provided by the user, who then may get a poor result depending on the URL provided. 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 9 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: TBD

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.