Aggregation Station

News Aggregation Algorithm - HackGT 7

Introduction

Technological innovation has put information at our fingertips. However, as the internet has grown and evolved, bad actors have begun to attempt to poison this new wealth of information.

Fake news is one of the largest influences of the spread of misinformation, especially within the last several years. It has caused increased tension between media outlets and the general public, undermining the legitimacy of all news and further dividing the public.Australia in particular has been under attack by Fake News, with a variety of sources. Careful balance between censorship and hindering the spread of misinformation. Especially in Australia, this has directly impacted the public trust in the Australian elections, including the most recent 2019 Federal election.
In response, the Australian government has taken steps such as the formation of the Electoral Integrity Assurance Taskforce, and Australian Electoral Commission among other committees (https://www.aph.gov.au/About_Parliament/Parliamentary_Departments/Parliamentary_Library/pubs/BriefingBook46p/FakeNews).

Our algorithm seeks to assist the Australian people by curating political news based upon a variety of factors, with the goal to eliminate all false, highly opinionated, or otherwise compromised, news. This is further explained within the following chapters.

Algorithm Design

For the purpose of this study, we will be considering “news” to be topically, socially, politically, and contextually relevant information regarding the happenings of a region- in our case, the region of Australia. Based upon this definition, we sought to provide at least one indication of each type of relevance. For topical considerations, we included the factor of “Age”: how long it has been since the article has been published. For social considerations, we included the factor of “Polarity”: how polarizing an article is and “Subjectivity”: how opinionated an article is. For political considerations, we included the factors of “Bias”: which way a publisher leans politically, “Credibility”: whether one can trust a news publisher. Finally, for contextual considerations, we included the factor of related “Tags”: what information was related to an article.

The Algorithm was built completely in Python and was formatted in an object oriented manner. A multitudes of libraries were used for a variety of different purposes, described as follows: Requests for API calls, MediaCloud to access a large collection websites and data corresponding

to those sites, Python-dotenv to protect our API keys, Textblob to evaluate subjectivity and polarity, and BeautifulSoup to webscrape. The specifics of the applications are further discussed within the "Factors" chapter.

Factors

Age

We considered this factor exponentially, as age largely relates to virality. Political news especially is punctuated by spurts of high interest in a given topic. For a topical American example, consider the Google trends graph for the term "Supreme Court" and "Black Lives Matter"- both graphs show an extreme spike followed by an almost immediate sharp decay. Based on these realities of political news, we used the function: $(x+1)^{-5x}$ to describe the appropriate weighting when considering the age of an article.

Bias and Credibility of the publisher

These statistics were both provided by https://mediabiasfactcheck.com/ and were obtained via the use of a python web scraper that employed the BeautifulSoup and Requests libraries. The website, mediabiasfactcheck.com, has already seen scholarly implementation within the U.S., including being used to create the University of Michigan's tool the "Iffy Quotient" and an MiT AI trained to fact check and the possible bias of a website.

Polarity and Subjectivity

These statistics were generated by the TextBlob library's sentiment analysis functionality (https://github.com/sloria/TextBlob), which, itself, extends NLTK (Natural Language Toolkit). Without going into too much detail, Textblob analyzes the parts of speech of a given text, evaluating the connotation of each term and pays special attention to the use of adjectives to return a float within [-1.0, 1.0] for polarity and a float within [0.0, 1.0] for subjectivity. In terms of polarity, a value of -1.0 represents solely one side of an argument and a value of 1.0 represents solely the other. Therefore, we concluded that taking the absolute value of this statistic would allow us to most choreantly apply it to the algorithm, with the large the absolute value, the poorer the url is ranked. In terms of subjectivity, 0.0 represents a completely objective article whereas 1.0 represents a completely subjective article.

Tags

The tags were obtained via the mediacloud API and represent common topics the article would have been associated with. The value of this factor was determined by considering the number and type of tags the article was associated with. Specifically, if the tags included "politics" the score would be dramatically increased and the longer the list of tags, the greater the score. Although we did consider immediately dismissing all of the articles not tagged with "politics", after some experimentation with the mediacloud API, we came to the realization that not all

political articles were tagged correctly by the API, and thus we decided only to “reward” the articles with the tags, rather than remove those without.

Weight Factors

Based on the considerations above, we determined the weights for each of these factors to be the following:

- Age: 9%
- Tags: 24%
- Polarity: 17%
- Subjectivity: 13%
- Credibility: 20%
- Bias: 17%

Program Output

Output as a .json file titled data_{current timestamp}.json with the top 55 ranked news articles according to our algorithm.

```
"1": #ranking{
 "title":
 "media_name":
 "author":
 "url":
 "metrics": {
   "age":
   "inlink_count":
   "outlink_count":
   "facebook_share_count":
   "tags": [
   ],
   "polarity":
   "subjectivity":
   "bias":
   "credibility":
   "liberal/conservative":
 },
```