Finding Bias in Political News and Blog Websites

Sonal Gupta

ABSTRACT

News and blog websites often have political bias (such as Republican, Democratic) in their articles. Automatic detection of the bias will improve personalized feed and categorization of news and blog articles. Our project aims to predict Republican vs. Democratic bias of news websites and political blogs using the phrases (a.k.a. memes) they quote in their text. We form a bipartite graph of websites and memes quoted by the websites. The algorithm starts with labels of a few websites and labels the rest of nodes in the bipartite graph using a simple label propagation algorithm. Our algorithm predicts labels better than supervised classification approach and other baselines.

1. INTRODUCTION

Politics oriented web documents and the links between them provide useful information about their content and the topics they span. However, finding political bias in the articles they publish is a hard problem because of the huge text corpus from the articles and sparse hyperlink information. An example would be to verify the commonly held belief that Fox News has Republican bias in their reporting. Automatic detection of political bias in websites can help in improving personalized feed of news articles and categorization of the websites.

Hyperlinks between the websites is an useful information for predicting bias of websites using their linking pattern. However, the hyperlink information is sparse and websites do not very often link to other websites. For example, they might write the quote stating the source but not explicitly linking to it. In addition, many of the websites may link to more prominent news sources, rather than the non-prominent ones. This may lead to higher concentration of in-links for famous websites and very small number of links to little known websites.

We can also approach the problem by considering full text of the articles, commonly known as sentiment analysis. Many of the sentiment analysis literature builds upon various NLP techniques like syntactic parsing, semantic parsing, negative and positive words dictionary lookup. However, sentiment analysis is still a very hard problem to solve, especially for a large corpus.

In this project, we propose a novel way of predicting political bias of websites using the phrases they quote in their articles. These quote phrases, or memes, are an useful source for predicting a website's bias. We believe that there are some discriminatory memes that are quoted by similar biased websites, and we can exploit the quoting pattern of memes to identify labels for websites. Table 1 shows memes with top chi-square score (see Chapter 13 in [6]) in the dataset. We can see that these memes have political inclination and are intuitively predictive of the websites bias that cite them more often. We build a bipartite graph of memes and websites, where each meme and website form a node and there is an edge between a meme node and website node if the website uses that meme phrase in its article. We start with a few labeled website node and find labels for other nodes in the graph. In this project, we show that a simple iterative algorithm that computes bias of websites and memes according to their neighbors in the bipartite graph, works well for the prediction task.

2. RELATED WORK

Research in sentiment analysis has largely been for finding opinion or sentiment in online reviews. Recently, people have also started looking at mining opinion in text of news articles and blogs. Adamic and Glance [1] studied the topics of discussion in online blogs and their linking patterns. They concentrated on blogs that either supported Republican or Democratic party. They found that the graph representing links between the blogs clearly has two big connected components suggesting that blogs with a particular political bias refer to similar biased blogs. They also analyzed the names associated with each of the blogs and found that blogs with Republican or Democratic bias refer more of Republican or Democratic people, respectively. The paper reveals interesting information about blogging bias on the Internet. However, it uses a lot of prior knowledge, like lists of the blogs and their labels (Republican/Democratic) from external websites, which might turn out to be very hard to do from text of the blogs. In addition, such external information is not available for most of the blogs and news articles. Also, they showed this phenomenon with only 40 blogs in total.

Meme	Chi-Square score
Joe the plumber	95.18
you can put lipstick on a pig	90.63
I think when you spread the wealth around it's good for everybody	89.51
yes we can yes we can	78.50
the chant is drill baby drill	70.00
I Barack Hussein Obama do solemnly swear	52.79
our opponent is someone who sees America it seems as being so imperfect imperfect enough that he's palling around with terrorists who would target their own country	51.81
not god bless America god damn America	51.81
the fundamentals of our economy are strong	49.87
I have protected the taxpayers by vetoing wasteful spending and championed reform to end the abuses of earmark spending by congress I told the congress thanks but no thanks for that bridge to nowhere	49.87
he is not spreading the wealth around	46.02

Table 1: Memes with top chi-Square distance in the dataset when 90% websites are labeled with their bias.

Godbole et al. [2] analysed sentiments of news and blogs on a large scale. They rely on manually created polarity lists and used WordNet for finding similar words. They then assign polarity of each entity in text and aggregates the polarity using statistical techniques. This work does not exploit any link information among the articles, and whether they appeared on same website or not. Matt Thomas et al. [7] investigated transcripts of U.S. Congressional floor debates and proposed an algorithm that uses sentiment analysis of discussions to predict whether or not the speech supports or opposes the proposed legislation. The system is useful when we have well formatted data with whole discussions in order to use the context to determine the speaker's inclination. This, however, is not the case with news and blog articles where is no 'discussion' among various people, and hence it is hard to make use of the context.

3. APPROACH

We generate a bipartite graph of memes and websites from our dataset. A meme node represents a meme in the dataset and a website node corresponds to a politically biased website. There is an edge between a meme node and a website node if the blog or news website used the meme in their text. We can either have directed edges from meme nodes to website nodes or vice-versa. In this project, we build the graph with directed edges from memes to websites. Given an initial set of labeled website nodes, our goal is to find bias of all the other website nodes in the graph. Let the graph be $G(V, E)$ where V is the set of vertices and E is the set of edges. Let $V_w \in V$ be the set of website nodes and $V_m \in V$ be the set of meme nodes. Let $V_w^l \in V_w$ be the initial labeled nodes. We assign weights (w_{dem}, w_{rep}) to each node to measure its bias towards Republicans and Democrats. Then, for each node not in V_w^l, we iteratively change the weights at each step depending upon the bias of its neighbors. The algorithm is described in Table 2.

The algorithm is similar to Hubs and Authorities approach suggested in [4] when we assume that memes are hubs and websites are authorities, and run the algorithm for both labels independently but normalize the scores at each node.

Note that as we are normalizing scores at node level, damp-

```
$t = 1$
while Num nodes in $V_w$ that change bias $> \epsilon$ do
    if $t$ is odd then
        $V_t = V_m$
    end if
    if $t$ is even then
        $V_t = V_w - V_w^l$
    end if
    for all $v_i \in V_t$ do
        $w_{dem}(v_i) = \sum_{v_j \in Ngh(v_i)} w_{dem}(v_j)$
        $w_{rep}(v_i) = \sum_{v_j \in Ngh(v_i)} w_{rep}(v_j)$
        $w_{dem}(v_i) = \frac{w_{dem}(v_i)}{w_{dem}(v_i) + w_{rep}(v_i)}$
        $w_{rep}(v_i) = \frac{w_{rep}(v_i)}{w_{dem}(v_i) + w_{rep}(v_i)}$
    end for
    $t = t + 1$
end while
```

Table 2: Psuedo-code of our algorithm. Roughly, at each iteration, democratic (or republican) score of a node is summation of democratic (or republican) scores of its neighbors. We normalize the scores at each node.

	Precision	Recall
Republican	80	15.4
Democratic	88.88	9.9

Table 3: Precision and recall for both classes when we use prediction of bias from the website name

ening factor and splitting bias according to node degree (see [3]) does not make any difference.

We also tried PageRank algorithm trimmed towards bipartite graph. We run the PageRank algorithm separately for both labels, and tried two approaches for normalizing scores. One is to normalize at each node so that each label weight is equivalent to probability, and the other is to normalize scores according to the node types (meme nodes and website nodes). They both give similar results.

As each meme acts as a feature, we tried selecting memes that are most useful for label prediction using their chi-square and mutual information scores (see [6]) computed by using labels of V_w^l. However, this either resulted in similar or worse performance than otherwise.

3.1 Predicting Bias from the Website Name

In our dataset, we noticed that many websites, especially blogs, have names that are predictive of their bias. For example, `http://stopbarackobama.com` gives a very strong idea about the bias of the website. We made a short list of Republican and Democratic words (approximately 10 each) and used a list of subjective words along with their polarity (for example, stop has negative polarity). We can then tokenize the website names in terms of words in our Republican/Democrat word list and the polarity list, and calculate polarity of the websites just on the basis on their names. Table 3 shows precision and recall for both classes when we use this approach to predict the bias. We can see that even though recall is low, precision is high.

4. EXPERIMENTS

4.1 Data Analysis

We used MemeTracker phrase cluster data by Leskovec et al. [5], which has meme clusters and the websites that used the memes in their text. We considered each cluster as one meme, and used only the blog websites in our evaluation. The bipartite graph we create from this data has 317600 nodes (246032 websites and 71568 memes) and 2564784 edges. For collecting blog labels we collected labels from `http://www.blogcatalog.com` and from Adamic and Glance [1]. There are in total 4086 labels from both sources but we have only 661 of those blog websites in our dataset. After pruning the original graph so as to keep only those websites nodes for which we have labels, the graph has 21887 nodes (661 websites and 21226 memes) and 55630 edges. We then remove nodes with degree 1 because they provide very less information about their inclination. The final graph we used has 16972 nodes (579 websites and 16393 memes) and 48858 edges. Out of 579 websites, there are 256 Republican and 323 Democratic websites. The degree distribution of the website nodes is in Figure 1(a) and the degree distribution of the meme nodes is shown in Figure 1(b). We can see that they follow power law.

To motivate that we can find bias of websites from this data, we define a metric *Average Difference Bias* (AvgDiffBias), which calculates if similar biased websites cite same memes. We define it as

$$\sum_{v\in V_m} \frac{|\text{Num Rep-Neighbors} - \text{Num Dem-Neighbors}|}{Degree(v)}$$

If there is some information or 'signal' in our graph, AvgDiffBias value of our graph should be higher than configuration model of the graph. We found that AvgDiffBias for our graph is 5704 as compared to 4301 for the configuration model. It shows that our graph has some information that can lead to accurate estimation of bias of the website blogs.

4.2 Methodology

To evaluate the performance of our algorithm, we perform a k-fold cross validation of predicting bias of websites nodes. We divide the website nodes in k folds in stratified manner, and in each fold we begin the reweighing algorithm with (k-1) folds of labeled data and evaluate the bias on the website nodes in the remaining fold. In our experiments, we use 10 fold cross validation. While predicting the final label of the website node, we say that the node is Republican if $w_{rep} > w_{dem}$ and vice versa. If both weights are equal, we randomly label the node. In Table 2, we chose ϵ as 20.

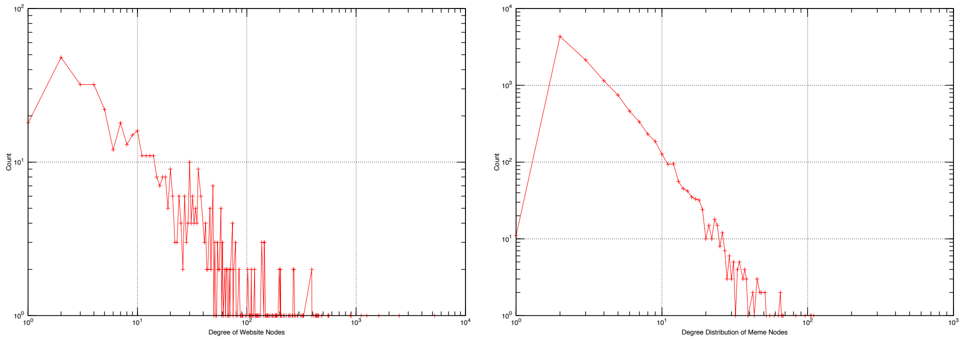
When we use the bias predicted by the website names, as described in Section 3.1, we label the test nodes with the prediction before running the algorithm as if the prediction by analyzing its URL is a true label. But the bias weights for these nodes can change in further iterations.

A random baseline is 50% since there are two classes to predict. Another baseline is to predict all nodes in test set by the label that has maximum number of nodes in the training set. We call it 'Max Baseline' in rest of the paper. We also compared our algorithm to supervised classification when we represent each website using 'Bag of Memes' and 'Bag of Words'. Bag of memes approach is similar to bag of words approach except that each website is represented as an unordered collection of memes instead of words. For bag of words approach, our word vocabulary was all the words contained in the memes. We use SVM implementation in Matlab for supervised classification approach.

4.3 Results

Our algorithm	80.63
Our algorithm with name prediction	82.15
PageRank	77.8
SVM using Bag of Memes	73.23
SVM using Bag of Words	66.56
Max baseline	55.49
Random Baseline	50.0

Table 4: Accuracy for different approaches to predict bias of blogs



(a) Degree of Website Nodes

(b) Degree of Meme Nodes

Figure 1: Degree distribution of web and meme nodes in the graph

The accuracy of predicting labels of website nodes by our algorithm, our algorithm with name prediction, and the baselines are shown in Table 4. We can see that our algorithm performs much better than the baselines. Also, using prediction of bias by exploiting polarity of words from the website names also helps in gaining better accuracy. Our algorithm also works better than supervised classification approach indicating the advantage of using the graph structure in predicting bias. We can also see that bag of memes approach works better than bag of words, may be because the words in the memes are not very informative.

Table 5 shows top Democratic memes and Table 6 shows top Republican memes predicted by our algorithm. We can see that the memes are mostly either supporting the corresponding party's presidential candidate or criticizing other party's candidate. We can also see that irrelevant memes that do not have anything to do with politics do not show up in the list. Table 7 show top Democratic (left side) and Republican (right side) blog websites predicted by our algorithm in one of the folds. Some of the names very predictive of their inclination, and most of them appear to be labeled correctly.

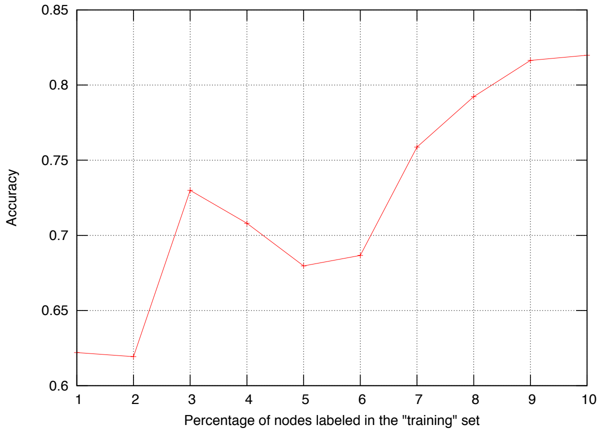


Figure 2: Variation of accuracy with increase in labeled nodes in the 'training' set

Figure 2 shows the variation of accuracy as we increase the number of labeled nodes in V_w^l (we call it training set though there is no machine learning in our algorithm) in the bipartite graph. The test set remains the same. As expected, we can see that more the number of labeled nodes, better the algorithm performs. Though, when only 10% of the nodes are labeled, it still performs better than the Max baseline.

4.4 Predicting Bias of Mainstream Media Websites

We can also predict bias of mainstream media websites using a similar approach as described before. However, in this case, we also include the mainstream media websites and the memes used in those website articles in our node set. As before, we begin with bias labels for a few blog nodes and get bias for rest of the nodes using the algorithm. Table 8 shows top republican and democratic mainstream media websites (whose degree is greater than 20) labeled by our algorithm in one of the k-folds (see Section 4.2).

CNN vs. Fox News

If we include famous mainstream media websites in our bipartite graph and run our algorithm, the democratic weight for CNN is predicted to be slightly more than Fox News (0.64 vs. 0.63 respectively). Though the difference is not much, it is somewhat similar to the perceived notion that CNN reporting is biased towards Democrats and Fox News reporting is biased towards Republicans.

5. REFERENCES

[1] Lada Adamic and Natalie Glance. The political blogosphere and the 2004 u.s. election: Divided they blog. In *In LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, 2005.

[2] Namrata Goldbole, Manjunath Srinivasaiah, and Steven Skiena. Large-scale sentiment analysis for news and blogs. In *ICWSM '07: Proceedings of International Conference on Web Social Mining*, 2007.

[3] Zoltán Gyöngyi, Hector Garcia-molina, and Jan Pedersen. Combating web spam with trustrank. In *In VLDB*, 2004.

[4] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5), 1999.

[5] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In

the way of the world a story of truth and hope in an age of extremism
you're looking at the miracle that John McCain helped create
she is a diva she takes no advice from anyone
decisions by the secretary pursuant to the authority of this act are non-reviewable and committed to agency discretion and may not be reviewed by any court of law or any administrative agency
I can't support the troops cuz every last one of them is being duped
tax and spend
it was a task from god
I'm John McCain and I approved this message
Pakistan is an ally in the global war on terror

Table 5: Memes with highest democratic scores assigned by our algorithm. We can see that the memes are mostly either supporting democratic agenda or focusing on criticizing Republican candidates.

when the stock market crashed Franklin D
he asked why we were not prepared to delay an agreement until after the US elections and the formation of a new administration in Washington
the breakthrough politics and race in the age of Obama
cbs evening news with Katie Couric
what I was suggesting you're absolutely right that John McCain has not talked about my muslim faith and you're absolutely right that that has not come
I've got two daughters 9 years old and 6 years old I am going to teach them first of all about values and morals but if they make a mistake I don't want them punished with a baby
we've got to have a civilian national security force that's just as powerful just as strong just as well-funded

Table 6: Memes with highest democratic scores assigned by our algorithm. We can see that the memes are mostly either supporting the Republican presidential candidate John McCain or criticizing the Democratic candidate.

donspoliticalblog.blogspot.com
thismodernworld.com
thesidetrack.blogspot.com
meaningfuldistractions.wordpress.com
thepoorman.net
thescottross.blogspot.com
conservativecat.com
robschumacher.blogspot.com
washingtonmonthly.com
theliberaloc.com

smarmycarny.com
hafezamohtar.wordpress.com
lecafepoliticien.blogspot.com
thedumbdemocrat.blogspot.com
rightwingchamp.com
realdebatewisconsin.blogspot.com
fafblog.blogspot.com
jeffblanco.blogdrive.com
afroamericanpieblog.wordpress.com
missbethsvictorydance.blogspot.com

Table 7: Top Democratic (on left) and Republican (on right) blog websites predicted by our algorithm in one of the k-folds.

swingstateproject.com
guntotingliberal.com
everydaycitizen.com
thecarpetbaggerreport.com
politicalcortex.com
elmundodeportivo.es
diariodecadiz.es
es.eurosport.yahoo.com

wizbangblog.com
bidinotto.journalspace.com
jewishworldreview.com
rightwingnews.com
mudvillegazette.com
americanthinker.com
espana-liberal.com
stoptheaclu.com

Table 8: Top Democratic (on left) and Republican (on right) mainstream media websites predicted by our algorithm.

KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 2009.

[6] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition, 2008.

[7] Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, 2006.