PAPER • OPEN ACCESS

Bias detection in Philippine political news articles using SentiWordNet and inverse reinforcement model

To cite this article: T A Quijote *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **482** 012036

IOP Conf. Series: Materials Science and Engineering 482 (2019) 012036 doi:10.1088/1757-899X/482/1/012036

Bias detection in Philippine political news articles using SentiWordNet and inverse reinforcement model

T A Quijote[1], A D Zamoras[2] and A Ceniza[3]

Department of Computer and Information Sciences, School of Arts and Sciences, University of San Carlos, Cebu City, Philippines

[1] thessaquijote@gmail.com, [2] allenazamoras@gmail.com, [3] amceniza@usc.edu.ph

Abstract. Not all information posted on the internet is deemed 'trustworthy.' Some articles, especially those related to politics, seem to display traces of bias, whether they be for or against the Philippine administration. This research aims to determine if a news article—and by extension, a news outlet—is biased based on its sentiments and use of lexica. Data were harvested from chosen websites and news outlets provided by Alexa. These data underwent pre-processing and were scored based on their sentiments with the use of SentiWordNet. The resulting scores were then fed into the Inverse Reinforcement Model, which determined whether an article is biased or not. With the use of Inquirer, Philstar, Manila Bulletin, The Manila Times, and Journal Online news articles, the system was able to detect bias with an accuracy rating of 0.89, precision of 1, recall of 0.60 and F-Measure of 0.75.

1. Introduction

Nowadays, more and more people turn to the internet as their primary source of news and information, but with this lies a problem: the internet is not as credible as some may think. Because of its vast and open nature, the internet is riddled with biased and deceitful content posted by non-credible sources, causing the spread of misinformation and pretenses.

According to Parker and Berman, bias means that observations are constantly influenced away from the truth and swayed towards one direction [1]. It poses a potential problem not only in clinical research, but also in articles of a political nature. Since people rely on news articles to deliver informative and accurate reports on current events, objectivity is a must when it comes to reporting the news.

It is important then to detect whether biases exist in the news articles we read, most especially if these biases present themselves subtly. Multiple solutions to this problem have been devised with the capacity to detect bias, especially with the use of sentiment analysis and its multiple approaches. In their comparative study, Collomb et al. [2] pointed out the differences among the approaches. The lexicon-based approach in particular is a famous one, and is known to work well and produce good accuracy ratings. This approach is usually implemented using lexicon detection, followed by sentiment strength measurement, such as context-based sentiment analysis. However, the integration of a different algorithm such as machine learning would achieve higher accuracy and boost its performance [3].

The employment of the Inverse Reinforcement Model, then, is unconventional. On its own, Inverse Reinforcement is more than capable of detecting bias and controversy [4], but combining it with lexicon-based sentiment analysis has not been documented before. The aim then is to detect bias in online news articles using SentiWordNet and Inverse Reinforcement.

2. Review of Related Literature

The following are related studies reviewed in this research.

2.1. Sentiment analysis in bias detection

In a study by Enevoldsen and Hansen [5], sentiment analysis was used to obtain more objective and quantitative measures of the political biases in newspapers. Their study focused on identifying how the two newspaper outlets, Berlingske and Information, portrayed two political parties, Alternativet and Liberal Alliance. The results showed that Information's articles about Alternativet were more positively worded than articles about Liberal Alliance, while Berlingske showed no significant difference in their portrayal of the two parties.

Another study by Zhang et al. [6] focused on the sentiment aspect of news articles to develop a system which detected and visualized sentiment tendencies of different websites. Their system would detect a website's sentiment bias when its tendency was different from the others. Further study using sentiment analysis to detect bias was conducted by Lazaridou and Krestel [7], wherein they aimed to detect the expressed sentiment an outlet will show about a politician which they captured in a signed graph.

2.2. Lexicon-based approach

Lexicon-based sentiment analysis is one of the most popular approaches to sentiment analysis. With this approach, a dictionary of words is required where each word is assigned a positive or negative sentiment value [8].

One study improved upon the traditional sentiment analysis (SA) approach—assigning text to a specific opinion category—by making use of the HK method. The main advantage of this method is the implementation of an automated analysis since it is a supervised SA. A dictionary has to be defined, though, which leads to its major drawback: the difficulty in classifying opinions expressed through ironic or paradoxical sentences, or in appreciating all the nuances of language, such as specific jargons [9].

2.3. Inverse reinforcement model in bias detection

The Inverse Reinforcement Model was developed in 2006 by Lauw, Lim, and Wang, where they observed that bias and controversy are mutually dependent, and proposed a reinforcement model to address this mutual dependency. They tested the model on product reviews where the reviewer assigned a 0 to 5 rating to an object. The reviewers were placed in ranked lists in descending order according to their bias values computed by Naive and Inverse Reinforcement. The results showed that there were significant differences between the ranked lists due to cases where Inverse Reinforcement disagreed with Naive by taking into account the mutual dependency between bias and controversy. These results were deemed to be encouraging [10].

Semine and Tumulak [4] developed a judging and scoring system in which the Inverse Reinforcement Model was embedded in order to detect bias and controversy. After a series of tests and validations, they concluded that the Inverse Reinforcement Model can be well implemented in an automated judging and scoring system.

3. Methodology

This section discusses the methodology employed by the research in order to develop a system that can effectively detect bias in Philippine news articles found online. Figure 1 illustrates the conceptual framework of the research.

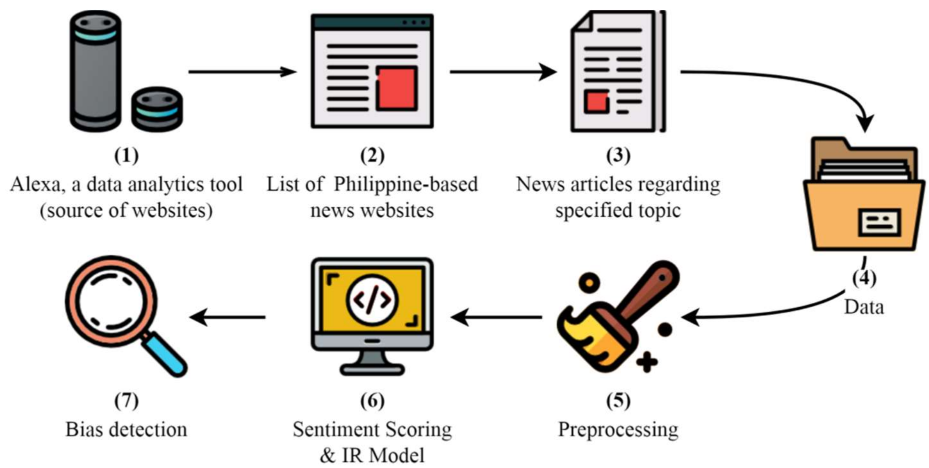

Figure 1. Conceptual framework of the research

3.1. *Gathering of data*
Alexa, a web traffic and data analytics tool from Alexa Internet, Inc., served as the source for this research (1). It ranked multiple news outlets in the Philippines based on popularity, and was used to identify which news websites should be considered to gather data from. Under the category 'News and Media > Newspapers', the news outlets used are: Inquirer.net, Philstar, Manila Bulletin, The Manila Times, and Journal Online (2).

Articles used for the research were manually picked out by the researchers. A topic was chosen, and articles related to the topic from each aforementioned news outlet was selected for scraping (3). A web scraper was used to extract the body of the articles, then exported into CSV files, which served as the data for this research (4). A total of 22,825 words were gathered.

3.2. *Pre-processing of data*
The collected data underwent pre-processing (5). Every character in the article was converted to lower case to obtain uniform casing. The resulting text was then tokenized into words. Stop-words such as "am", "the", "she", "to", etc. were removed from the document. All punctuation marks, except for the hyphen, were then removed from the remaining text. 13,126 words were left for processing.

3.3. *Scoring with SentiWordNet*
After pre-processing, all words in the document were assigned a raw score (6). SentiWordNet, a lexical resource for supporting sentiment analysis, was used to assign three scores: positivity, negativity, and objectivity score. The assigned raw scores were within the range 0 to 1, where the sum of the scores was 1 [11]. The highest score was identified and served as the word's raw score.

Afterwards, each document was classified into either positive or negative [12]. The total score for the positive and negative words were calculated separately, then normalized by calculating the ratio of each score to the sum of both scores. If the normalized positive score was greater than the normalized negative score then the document was labeled positive. Otherwise, negative.

3.4. *Inverse reinforcement model*
The Inverse Reinforcement Model was employed to detect bias (6), which adopted an iterative process. Initial data were taken from the tabulated scores produced from sentiment scoring. The model then computed for the deviation values for each outlet on each article they published using the formula for deviation, Equation 1. d_{ij} is the average distance between r_i's score and the score of each co-outlet r_k, where $m_j > 1$ denotes the number of co-outlets, e_{kj} is the score of r_k, and e_{ij} is the score of r_i.

$$d_{ij} = \frac{1}{m_j - 1} \sum_{r_k \neq r_i} \left| e_{kj} - e_{ij} \right| \tag{1}$$

$$b_i = Avg_j \, d_{ij} \cdot \left(1 - c_j\right) = \frac{\sum_{j=1}^{n} d_{ij} \cdot (1 - c_j)}{n_i} \quad (2)$$

$$c_j = Avg_i \, d_{ij} \cdot (1 - b_i) = \frac{\sum_{i=1}^{m} d_{ij} \cdot (1 - b_i)}{m_j} \quad (3)$$

n is the total number of articles, m is the total number of outlets, n_i is the number of articles evaluated by r_i, and m_j is the number of outlets evaluating article o_j.

From the computed deviation values, the initial bias values b_i were calculated using the formula in Equation 2, with the average deviation values for each outlet as the initial controversy values c_j. The controversy values c_j were then computed using the formula in Equation 3, taking the computed initial bias values b_i and the deviation values as inputs. The bias values were computed again using the formula in Equation 2, now using the deviation values and the newly computed controversy values as inputs. The whole process of computing the bias values and controversy values were repeated using each other's values and deviation values as inputs, thus, applying an iterative process. The iterations stopped only if they reached convergence, which was when the bias values do not change [10].

The Inverse Reinforcement Model assumed that all outlets are biased, but differed in their level. Bias levels were categorized into four according to the level of bias which covered outlets, as shown in Table 1.

Table 1. Four levels of bias.

Category	Bias Values	Level of Bias
Level 1	Lower than 0.1	Allowable bias
Level 2	Above 0.1 but lower than 0.2	Relatively considerable bias
Level 3	Above 0.2 but less than 0.25	Biased
Level 4	Above 0.25	Extremely biased

To properly detect bias, the system considered an allowable bias of not more than 0.19 (7). This allowed the system to detect relatively biased outlets and articles [4].

4. Results

Before feeding the actual gathered data into the system, testing was conducted to verify the validity of its results. The system was first tested on synthetic data wherein an outlet's articles simulated bias sentiments. The second test involved analyzing unbiased articles. The third test was a combination of biased and unbiased articles to test the application's accuracy in detecting both. After passing all three tests, the application was used to determine bias in actual data. The system's accuracy, precision, recall, and F1 score were calculated based on the following formulas:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 \cdot (Recall \cdot Precision)}{(Recall + Precision)} \quad (7)$$

where TP is the number of correctly predicted biased articles, TN is the number of correctly predicted unbiased articles, FP is the number of wrongly predicted unbiased articles, and FN is the number of wrongly predicted biased articles. A threshold of 80% served as the basis for success.

Table 2. Test 1 with simulated bias sentiments.

Outlet	Sentiment Values[a]	Deviation Values[a]	Bias Values
1	0.44	0.12	0.10
2	0.44	0.13	0.11
3	0.41	0.14	0.11
4	0.43	0.12	0.10
5	0.85	0.42	0.35

[a] Values shown are averages of the values of the 'articles' in each outlet.

For test 1, outlet 5 was purposely given biased scores. Results in Table 2 showed that outlet 5 was correctly predicted as biased, while the rest were correctly predicted as unbiased.

Table 3. Test 2 (a) with unbiased articles and Test 3 (b) with a combination of biased and unbiased articles.

(a)				(b)			
Article	Sentiment Values	Deviation Values	Bias Values	Article	Sentiment Values	Deviation Values	Bias Values
1	0.16	0.20	0.17	**1**	0.20	0.32	0.27
2	-0.17	0.19	0.16	**2**	-0.17	0.16	0.14
3	0.17	0.21	0.18	**3**	-0.22	0.19	0.15
4	-0.13	0.15	0.13	**4**	-0.27	0.23	0.19
5	0.14	0.19	0.16	**5**	0.17	0.29	0.24
6	-0.11	0.15	0.12	**6**	-0.11	0.15	0.13
7	-0.11	0.15	0.12	**7**	-0.11	0.15	0.13

The system correctly predicted all unbiased inputs as unbiased for the second test. For the third test, articles 1 and 5 were correctly predicted as biased; 2, 6 and 7 were correctly predicted as unbiased; and articles 3 and 4 were falsely predicted as unbiased. Combining all TP, TN, FP, and FN values across all three tests yielded an accuracy of 0.89, precision of 1, 0.60 recall, and 0.75 F1.

Table 4 presents the results produced by the system after feeding it with the actual gathered data.

Table 4. Results of the bias detection system with actual data.

News Outlet	Sentiment Values[a]	Deviation Values[a]	Bias Values
Inquirer	-0.14	0.19	0.16
Philstar	-0.18	0.15	0.13
Manila Bulletin	-0.05	0.22	0.18
The Manila Times	-0.08	0.23	0.19
Journal Online	-0.10	0.18	0.15

[a] Values shown are averages of the values of the articles in each outlet.

According to the results, articles from The Manila Times were considered to be biased, and by extension, the outlet as well.

5. Conclusions

This research was able to detect bias from online news articles by implementing the Inverse Reinforcement Model with SentiWordNet. Using SentiWordNet, the general sentiment of articles were obtained and were later used as input for the Inverse Reinforcement Model. The model calculated the deviation values and controversy values of the articles to determine the bias values of the outlets. After a series of tests and validations, the system was able to determine that articles from leading Philippine-based news outlets were unbiased, except for those from the Manila Times. The Manila Times was then deemed to be biased, with an accuracy rating of 89%.

The results produced by the research are known to be significant since they provide concrete values and figures–i.e. accuracy, precision, recall, and F1–that describe how successful the system is. Other related studies and approaches have failed to state how successful their methods were in terms of statistical approaches. In addition, this research focused on the news articles themselves and how effective the IR Model was on them, seeing as how it was not used on this type of data before. Consequently, this research did not dwell on the political parties that news outlets sided with.

It is important to note that the current system only determines the magnitude of bias. For our future work, we wish to include identifying the direction of bias to further enhance our system's capability. Furthermore, we would also like to consider the incorporation of other sentiment analysis techniques.

6. Acknowledgements

The researchers would like to extend their gratitude to the Department of Computer and Information Sciences and the University of San Carlos for supporting this research endeavour.

References

[1] Parker R A and Berman N G 2016 *Planning Clinical Research* (Cambridge University Press) pp 203–16
[2] Collomb A, Costea C, Joyeux D, Hasan O and Brunie L 2014 A Study and Comparison of Sentiment Analysis Methods for Reputation Evaluation
[3] Mudinas A, Zhang D and Levene M 2012 Combining lexicon and learning based approaches for concept-level sentiment analysis *Proc. of the First Int. Workshop on Issues of Sentiment Discovery And Opinion Mining* p 5
[4] Semine R V and Tumulak J 2010 Automated bias detection judging and scoring system using inverse reinforcement (IR) model *Philippine Information Technol. J.* vol **4** no **1** pp 3-10
[5] Enevoldsen K C and Hansen L 2017 Analysing political biases in danish newspapers using sentiment analysis *J. of Language Works* **2(2)**
[6] Zhang J, Kawai Y, Nakajima S, Matsumoto Y and Tanaka K 2011 Sentiment bias detection in support of news credibility judgment *44th Hawaii Int. Conf. on System Sciences* pp 1–10
[7] Lazaridou K and Krestel R 2016 Identifying political bias in news articles *Bulletin of the IEEE TCDL* **12**
[8] Jurek A, Mulvenna M D and Bi Y 2015 Improved lexicon-based sentiment analysis for social media analytics *Security Informatics* **4(1)**
[9] Ceron A, Curini L, Iacus S M and Porro G 2014 Every tweet counts? how sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to italy and france *New Media & Society* **16(2)** pp 340–58
[10] Lauw H W, Lim E and Wang K 2006 Bias and controversy: beyond the statistical deviation *Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* pp 625–30
[11] Baccianella S, Esuli A and Sebastiani F 2010 Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining *Lrec* **10** 2010 pp 2200–4

[12] Sharma S, Chakraverty S, Sharma A and Kaur J 2017 A context-based algorithm for sentiment analysis *Int. J. Computational Vision and Robotics* **7**