Political Ideology Detection Using BERT and Convolution Neural Networks

Ben Sukboontip

Abstract

News articles have been more biased than ever with the integration of the internet in our daily lives. With a lack of official tools and man-power to detect these biases, several autonomous models have been proposed in the past that aim to detect political ideologies of news articles using natural language processing methods. Some works such as (Dayanik & Padó, 2020) aim to mask actors to reduce frequency biases of models while other papers such as (Kulkarni et al., 2018) aim to incorporate multiple information other than the news contents themselves to compute political ideologies of news articles. This implementation aims to show that information aside from news articles themselves can aid the models in computing biases of articles. More specifically, the titles of news articles were encoded using Bidirectional Encoder Representations from Transformers (BERT) proposed by (Devlin et al., 2018) and subsequently inputted into a pre-trained BERT model and multiple convolutional neural network layers to classify news articles as left, right, or center leaning. The implementation has shown promising results, proving that information aside from the content of articles themselves can help models learn more accurately.

1. Introduction

News media has caused more political polarization among people than ever before. While the basis of journalism is to remain unbiased, there will always be some amount of biases in news articles whether it is intentional or not. Therefore, most news outlets write articles biased towards their own political agenda that ultimately culminates in biased news. Moreover, with the customized nature of search engines and social media, people biased towards one political side will likely only receive news biased towards them. Therefore, having a way to detect biases among news articles spreading around the internet to inform readers of pre-existing biases has never been more paramount. However, due to the infeasibility of manually detecting the biases of all articles across all platforms simply due to the copious amounts of texts in existence, an autonomous way of detecting political ideologies or bias seems like the most viable approach.

In the past, there have been many different natural language processing (NLP) approaches to this problem. One way is to focus on the sentence-level structure of texts, using the relationships between words within a sentence as information to train the model, while others focus on the word-level structures of texts, ignoring the relationships between words within a sentence and view each word as their own entity when designing their implementations and approaches. Another common method is to also use the author's information to determine the characteristics of the text as they view each author as having their own implicit characteristics, regardless of if it is conscious or subconscious.

While other previous and related works have similar implementation as this, none has even attempted the multi-view approach in (Kulkarni et al., 2018). The paper has proposed a multi-view model for political ideology detection that ultimately combines three main aspects of news articles: title, content, and reference articles. The paper computes the results by training each element of the paper individually and then concatenating the results together to be inputted into a linear layer followed by ReLU to assure non-linearity of the model and then another linear layer to classify the output as left, right, or center leaning.

Due to the lack of reference code relating to related papers on this implementation, this paper aims to re-implement (Kulkarni et al., 2018) by detecting the political ideologies of news articles based on their titles alone. Using NLP methods to encode the title, a string format, into tensors to input the following tensors into multiple convolutional neural network (CNN) layers as CNN has been shown by this paper, (Wang et al., 2018), and (Dayanik & Padó, 2020) to be effective in evaluating short texts such as titles and microblog posts. By re-implementing only the title aspect of (Kulkarni et al., 2018), this paper aims to show that titles can hold enough content to be able to accurately classify political leanings alone without the need for the contents of the news articles themselves, further supporting the results shown by the multi-view approach.

2. Related Works

This section explains three research implementation related to this paper. The first related work explains the multi-view approach this paper aims to support. The second related work utilizes the existing BERT model to assess the impact of the censorship of author information to reduce the inherent frequency bias of machine learning models. Lastly, the third related work uses a user-attention-based CNN to compute the sentiment classification of microblog posts across two different languages.

2.1. Multi-view Models for Political Ideology Detection of News Articles

The main purpose of (Kulkarni et al., 2018) is to present a multi-view model that could detect the political ideology of a news article. The paper firmly believes that there are inherent biases in news articles despite the fact that news should fundamentally remain unbiased. However, manually labeling the political ideologies of all news articles circulating on the internet is impossible given the large amounts of articles circulating online, so the paper offers a way to autonomously detect political biases without human labor. Specifically, this paper differs from other research in this field in that this paper aims to detect biases on the news article levels whereas previous works have focused more on detecting biases on the sentence level. By detecting biases on the news article level, the paper was able to add more features to train the model compared to when detecting biases on a smaller scale which focuses solely on the texts and occasionally the author. This paper aims to combine three main features of a news article input: the title, the links to other news media sources excluding the links to the author's other articles, and the content itself. In terms of the data set, this paper utilizes the publicly available information on AllSides.com, a website containing over 59 US-based news sources with their respective political ideologies, by extracting the title, the cleaned content, and the external links within the article to train and test the model.

Based on the input, we can view the input of the model as follows:

$$X = \{X_{title}, X_{net}, X_{content}\}$$

$$Y = \{LEFT, CENTER, RIGHT\}$$

In the above equation, X would be the input of the model and Y would be the output. The model proposed by the paper would ultimately calculate the probability of Y given X. The process is then divided into three main parts: the discriminator, the approximate posterior, and the prior. The purpose of the discriminator is to define the differences between the model's prediction and the true labels taken from the data set. Because of the simple nature of the discriminator, the paper uses a feed-forward neural network model for the discriminator. Next, the approximate posterior uses an inference network that divides the three inputs to train each input with different neural network models. In this case, it uses CNN for the title because CNN has shown to work extremely well with short texts such as titles. The paper then uses a Node2Vec implementation, viewing each link to external sites as nodes, to model the network structure of articles. Lastly, the content is modeled using a hierarchical approach, using the content both on the word level and sentence level. The output of the three inputs is then concatenated through a linear layer as shown in Figure 1 so that it outputs two main components: the mean and log-variance. In terms of the prior, the paper uses the Gaussian distribution with diagonal co-variances but states that many different implementations of priors could be incorporated. When the model is compared to other recent works, the proposed model outperforms its best competitor by approximately 10%.

The paper tested its model three different times. The first time, the paper uses the title and links as inputs. The second time, it uses the title and content as inputs. The third time, it uses all three components of the proposed inputs as inputs. By assessing articles in the new article-level rather than the sentence-level, the main strength of the paper is its utilization of additional features that many models fail to neglect. Interestingly, the first two iterations of the test yield very similar results, which are similar to many other models based on recent works. The important point to note is that in the first iteration, the content of the news article was never used as input. Before the experiment, the content should seemingly play the most important role in determining the political ideology of the paper. However, the results of the experiment in the paper suggest that the title, content, and links to external sources all play important roles in detecting a precise output. Moreover, the paper observes that experiments where only the contents and titles were used as features only slightly outperform the baseline while experiments that specifically included network cues perform much better than baseline. Because of this, the paper suggests that having references have unique cues that contents and titles do not possess. Because of this basis, this paper attempts to further this point by attempting to classify the political ideologies of news articles solely using just the title as input features. Regardless, this does not mean that the model does not have weaknesses. The paper fails to accurately determine the political ideologies of non-political articles such as articles related to entertainment or education. Lastly, one can argue if the results of the experiments could be better if better algorithms were used for some parts of the model. The model uses several feed-forward neural networks, a very simple type of neural network. If these models could be replaced with its descendent, the recurrent neural network, or even a deep neural network, could the

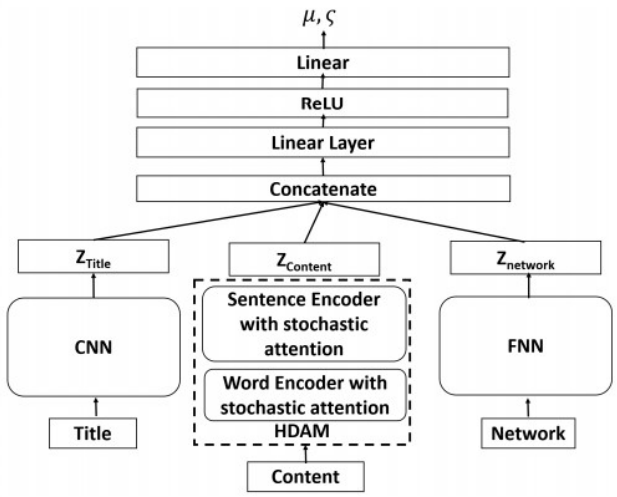

Figure 1. Topology of multi-view model

output of the experiment be improved?

2.2. Masking Actor Information Leads to Fairer Political Claims Detection

In (Dayanik & Padó, 2020), the paper claims that frequency bias affects almost all recent and related works in the field of Computational Social Sciences (CSS), a field founded on fairness and lack of bias. Frequency bias in this paper refers to the fact that most models are able to detect and recognize texts from actors, the author of a claim, that appear often in the training set. Therefore, most of the state-of-the-art works fall prey to the frequency bias, learning the actor and their general stance rather than the political claims of the message itself. This means that actors who make repeated claims will cause models to falsely give higher certainties to these political claims without much proof. This phenomenon can be viewed as a type of over-fitting as it incorrectly fits political claims to the author rather than the text itself. Consequently, the paper proposes a debiasing method that censors the actors and specific pronouns to reduce the bias and allow to model to objectively detect the political claims of texts.

In terms of the dataset, the paper uses a corpus of articles related to a 2015 debate on German immigration. All of the data set is drawn from a German newspaper Die Tageszeitung. The paper proposes and experiments with two types of censorship or masking techniques: MaskName and MaskNamePron. MaskName aims to masks all names of persons inside the texts so that the model could not utilize the actor information to determine the political claim of a text. MaskNamePron is a stricter version of MaskName where in addition to the masking done in MaskName, MaskNamePron also masks all types of personal pronouns in the text, eliminating all possible ways that the model can detect the actor and use that information to determine the text's political claim. When the political claims are masked, the paper then inputs the texts into a widely successful political claims detection algorithm proposed in (Padó et al., 2019). However, because the input data set is in German, the paper uses the BERT model trained specifically with the German corpus instead of a BERT model trained with multiple languages due to complications or inaccuracies that could arise from training a multi-lingual model. After experimentation with this model, the paper has shown that masking the actor's information greatly reduces frequency bias while still maintaining the accuracy and precision of previously proposed models. Moreover, it has also shown that the MaskNamePron masking slightly outperforms MaskName masking in all aspects.

While it is clear is the frequency bias is common in most proposed models and research in CSS, it is important to question how important debiasing the frequency bias is. This is because the output of the masked model is extremely similar to that of the unmasked model. In fact, (Wang et al., 2018) proposes that having actor information can actually help in determining the sentiment of a text. Despite the fact that political claims detection and sentiment classification are inherently different problems, the inclusion or exclusion of actors should have the same effect on the result. Both papers contradict each other regarding the existence of authors or actors in the output. Moreover, one can argue that if the model uses CNN to detect political claims of short texts instead of its current method, better results could have been produced as more sophisticated and modern models are used. This is because the other two papers have championed the effectiveness of CNN when dealing with small texts. However, we can understand that the goal of this paper is not to propose a new political claims detection method but to prove that masking the actors' information does not reduce the effectiveness of the model.

2.3. Personalized Microblog Sentiment Classification via Adversarial Cross-lingual Multi-task Learning

(Wang et al., 2018) aims to classify sentiments of microblog posts by heavily utilizing information on the author of the post. Therefore, the paper rests on the assumption that all authors of microblog posts have innate biases and opinions that is consistent in all posts. The paper posits that personal distinctions such as language habits, personal character, roles, and even political views can be used to enhance the results of sentiment analysis on microblog posts. Because of this consistency of personal views, the paper can build parameters around authors to help determine the sentiments of their posts. For data sets, data is crawled from two popular microblogs: Twitter and Weibo. However, one distinction

that should be noted is that Twitter is mainly an English microblog which Weibo is mostly in Chinese. Regardless, the observations mentioned in the paper has shown that personal views and opinions are consistent with authors of both English and Chinese microblog posts. The paper mainly uses CNN for the model because CNN has been shown to work well with short texts. This application of CNN has also appeared in (Kulkarni et al., 2018), where the CNN model is incorporated into the model taking titles as inputs because if its effectiveness with small texts.

The methodology of the paper is mainly divided into two parts. Firstly, the model must capture the individuality of different authors of microblogs to be able to utilize the uniqueness of authors in determining the posts' sentiments. To do this, the paper uses the attention mechanism to classify the authors. Secondly, CNN is used to incorporate the authors' parameters into different posts and enhance the representation of microblog posts to give a more accurate sentiment classification. Ultimately, the post is classified as either having a positive or negative sentiment. This model was then tested and compared with several tried-and-true methods for microblog sentiment classification. Despite the model besting other baseline models, the increase in performance is less than 3%.

While the paper succeeds in showing that individuality can help in classifying if a post is positive or negative, it begs to question if the author truly plays a role in the sentiment of microblog posts. People vary in emotions all the time and it is very common for positive and negative posts to come from the same authors. It is uncertain if the assumption that people have innate individuality takes into account the fact that the authors' current emotions, which are often turbulent, also massively determines the sentiment of microblog posts. Therefore, it seems as if utilizing the author for microblog sentiment classification can be viewed as both a strength and a weakness depending on one's point of view.

3. Implementation

3.1. Dataset Construction

This paper utilizes Kaggle's "All the news" dataset which contains 143,000 news articles from 15 American publications. This dataset is different from the one (Kulkarni et al., 2018) uses in that this consists of only 15 American publications while that paper uses a dataset that contains 59 US-based news sources. Another difference between the two implementations is that the other paper writes its own code to spider several news sources from AllSides.com while the implementation in this paper simply loads its dataset from Kaggle as CSV files. The initial dataset includes a variety of information about each article such as the article title, publication name, author name, date of publication, year of publication, month of publication, URL, and article content. However, for the implementation of this paper, only the title and the publication name was used. The publication names of each article were cross-referenced with their political leaning rating on AllSides.com in order to obtain the political leanings of each article so that supervised learning could be done. Using this method, we can determine, for example, that articles from Reuters are center-leaning, articles from Fox News is right-leaning, and articles from Vox is left-leaning. Next, the dataset is then cleaned so that it excludes rows where information is corrupted. Furthermore, the texts of the titles were further cleaned using regular expression to exclude complicated punctuations that are not question marks while also removing special characters among other operations. Lastly, the articles' political leanings were encoded into a numerical format where 0 represents left-leaning, 1 represents center-leaning, and 2 represents right-leaning. However, due to the naturally biased nature of news articles, there are many more left and right-leaning articles than center-leaning articles as shown in Figure 2. This crucial distinction could potentially come into play when scrutinizing the results of this paper.

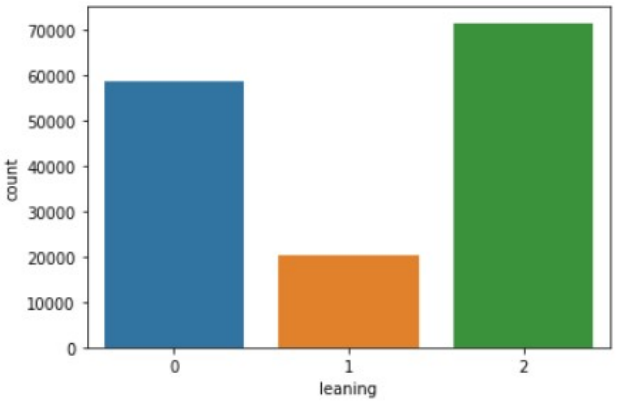

Figure 2. Dataset composition

3.2. Text Encoding

Because this paper plans to utilize CNN to classify the political leanings of the news articles by using their titles, we must encode the titles to represent them as tensors, a form readable by the CNN, rather than texts which are unreadable to the CNN. This implementation uses the encoder used in BERT to represent the titles as tensors. BERT is a pre-trained attention-based model developed by Google that utilizes the concept of bi-directional or directionless learning rather than other popular models at its time such as Long Short-Term Memory (LSTM) models which only analyze data either from left-to-right or right-to-left. Because BERT is bi-directional, it is able to look at contextual relationships

between words before and after it, allowing the model to look at the sentence as a whole instead of reading it word by word, allowing for better results as contexts were also used.

Firstly, every unique word in the corpus, the combination of all the titles, is embedded so that each word is represented with a unique vector that is uniform in size n. Because the model is pre-trained, the vector representation of each word has already been computed so no computation power or computation time is used in this paper. Next, every title is tokenized so that the words are represented as individual elements inside a list. The tokenized titles are then numericalized to represent the indices of each word inside the word corpus so that each unique word can be represented with a unique number that maps it to the word vector in the corpus. To assure that all the title sizes are uniform, all title vectors are padded during tokenization to a number greater than the size of the longest title. In this case, all titles were padded to a length of 40. Therefore, each title can be represented with a matrix of size 40 x n where 40 represents the size of each word in the embedding and n represents the length of the vectors that represent each word according to the pre-trained BERT model. To ensure that the padding will not interfere with the learning process, the attention mask is used to tell the model which token will be used and which token will not be used. Attention masks will be a matrix of size 40 x n, where each element in the matrix is represented with either a 0 or a 1, telling the model to use this specific token if the corresponding attention mask is 1 and telling the model the token is part of the padding if the corresponding attention mask is 0.

3.3. Fine-Tuning BERT

In this paper, transfer learning is utilized on the BERT model so that the pre-trained BERT model is adjusted to the scope of this project. In the pre-trained BERT model, (Devlin et al., 2018) leverages two main processes to train the BERT model. Firstly, BERT uses a process called Masked Language Model (MLM) that masks 15% of all words in its WordPiece corpus. This is done because the nature of bi-directional computation is that the word can ”see” itself allowing the model to trivially predict its own word. Therefore, MLM is used to create a bi-directional pre-trained model. Secondly, BERT also uses a process called Next Sentence Prediction (NSP) to train the model to predict which sentence comes first given a pair of sentences. The paper has shown that his process greatly increases accuracy in tasks such as Question Answering and Natural Language Inference.

The BERT pre-trained model can be used in many varieties of tasks by fine-tuning the model by inserting a classification layer at the end of the pre-trained BERT model. In this case, a dropout layer is added to regularize the output of the pre-trained BERT model and then inputted into a fully-connected linear layer with an output feature of 3, representing the three possible political leaning classifications of each news article. Here, the final linear layer acts as a classification layer that allows the output to be represented with 3 features. In addition to the above layers, this paper also implemented three convolutional layers between the pre-trained BERT model and the dropout layer for additional learning. The model is then trained across 5 epochs with a batch size of 8 to achieve the results explored in the next section.

4. Evaluation and Results

For this paper, the pre-processed data is split into three different groups: train, test, and validation groups. The initial dataset is split into two sets, the train and the test sets so that the training group consists of 90% of all datasets. After that, the test set is then split equally into the validation set and the test set. Initially, the full dataset consisting of 143,000 news articles is trained in the model to achieve initial results. However, it should be noted that the amount of center-leaning articles in the dataset is significantly less than the left and right-leaning articles which can create an imbalance of the training data. Figure 2 shows that the number of center-leaning articles is approximately 3 times less than the number of left or right-leaning articles. This distinction can cause the model to learn with a bias against center leaning models, something that could affect the accuracy of this model. As shown in Figure 3, some of the errors in the confusion matrix are related to the center-leaning articles where the model incorrectly predicts a center-leaning article is something else or incorrectly predicts other articles as center-leaning. However, one interesting thing that should be noted in this experiment's results is that the model seems to be able to detect right-leaning articles more accurately. This success can most likely be attributed to the fact that a large part of the dataset consists of right-leaning articles, being 15% more than left-leaning articles, and 250% more than center-leaning articles. Another factor that could cause these errors could be the fact that many of the right-leaning articles in the dataset are from extremely right-leaning publications such as National Review, Fox News, and Blaze while the left-leaning articles in the dataset are from slightly left-leaning publications such as The Guardian, New York Times, and The Atlantic.

The counter this, another experiment is conducted where the model was trained with a dataset where the left and right-leaning articles were discarded so that the number of left, right, and center leaning articles are uniform across the entire dataset. In Figure 4, the results show that truncating the dataset does not have a significant impact on the precision of the model. The number of incorrectly predicted

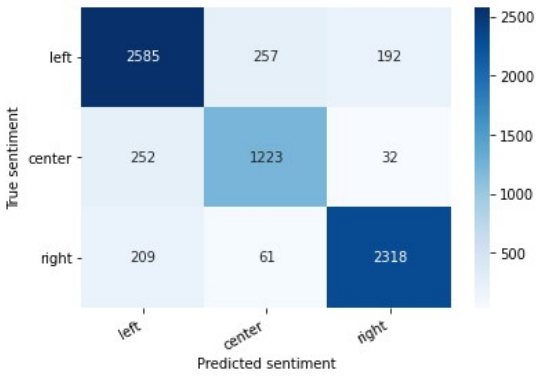

Figure 3. Confusion matrix for entire dataset

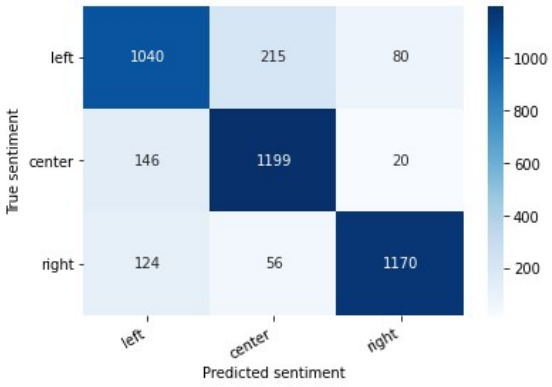

Figure 4. Confusion matrix for truncated dataset

	Precision	Recall	F-Score
Left	0.85	0.85	0.85
Right	0.79	0.81	0.80
Center	0.91	0.90	0.90

Table 1. Accuracy for first model

	Precision	Recall	F-Score
Left	0.79	0.78	0.79
Right	0.82	0.88	0.85
Center	0.91	0.87	0.89

Table 2. Accuracy for second model

center-leaning articles are similar compared to the first experiment's. However, Table 1 and Table 2 shows that when training with the entire dataset, the model is able to predict left-leaning articles slightly more accurately but predict right-leaning articles less accurately compared to the model that is trained with the truncated dataset. Moreover, the types of errors for both experiments are relatively similar in terms of the distribution of the types of errors. When comparing the precision, recall, and F-score of the two experiments, we can conclude that both experiments perform extremely well.

One possible reason why the model worked extremely well is that according to (Dayanik & Padó, 2020), sometimes, the model focuses on the author itself rather than the content, the title in this case. This is especially significant because the left, right, and center leanings of news articles in the dataset are determined by the publications. Therefore, it is also possible that the model is actually predicting the political leanings of news articles by using inherent characteristics of the author or the publications to classify the articles. Another point that should be noted is that one possible reason for the extremely accurate model is because of the pre-trained BERT model that has been fine-tuned for the classification specific to this paper rather than due to implementation of multiple CNN layers as CNN alone should not be able to create results this accurately. Therefore, with the help of the pre-trained BERT mode, we can conclude that the titles alone are sufficient in extracting the political-leanings of news articles.

maybe talk about how authors cause political bias

5. Conlcusion

In this paper, we propose that news article titles alone can be used for political ideology detection of news articles through a pre-trained BERT model and multiple CNN layers. We show that the titles alone can accurately predict the political ideologies of news articles without the help of the content itself, suggesting that titles are relevant as features in training models to detect political leanings.

References

Dayanik, E. and Padó, S. Masking actor information leads to fairer political claims detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.acl-main.404.pdf.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Kulkarni, V., Ye, J., Skiena, S., and Wang, W. Y. Multi-view models for political ideology detection of news articles. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1388.pdf.

Padó, S., Blessing, A., Blokker, N., Dayanik, E., Haunss, S., and Kuhn, J. Who sides with whom? towards computational construction of discourse networks for political debates. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2841–2847, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1273. URL https://www.aclweb.org/anthology/P19-1273.

Wang, W., Feng, S., Gao, W., Wang, D., and Zhang, Y. Personalized microblog sentiment classification via adversarial cross-lingual multi-task learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1031.pdf.