Erasmus University Rotterdam

Erasmus School of Economics

Bachelor's Thesis BSc^2 Econometrics and Economics

Classifying hyper-partisan news: Using linguistic analysis to predict the political orientation of news articles

Max Groeneweg (464534)

Supervisor: Professor Aurélien Baillon

Second Assesor: Professor Lumsdaine

July 2, 2021

Abstract

In recent years, social media has come under repeated scrutiny regarding its responsibility in regulating the news that circulates on its platform. With false and misleading content to be found throughout the internet, the need for automatic detection of this content is becoming increasingly important. This paper seeks to use linguistic analysis, through the software LIWC, to build a model which can correctly identify whether an article is of a left or right wing nature as well as identify its veracity. Veracity proved difficult to predict with linguistic variables but promising results are found for classifying political orientation. Logistic Regression, Decision Tree Regression and eXtreme Gradient Boosting Regression are implemented to predict the political orientation of articles. All three models outperform random guessing. Using eXtreme Gradient Boosting an out-of-sample accuracy of 76% is achieved, with the Logistic Regression and Decision Tree Regression following closely.

Acknowledgements

I would like to thank Professor Baillon for his guidance and insights throughout the thesis process. His excitement when discussing the research topic was a large driving factor behind the work in this paper. His optimism and solution driven attitude when the research results first presented its challenges showed me that research difficulties should not stand in the way of an interesting thesis and how there is always a way to work with the results you are given.

Contents

1 Introduction

The 2016 United States presidential election was a tumultuous period leading to, perhaps, one of the most interesting developments in political strategy seen in the last decade. While the election was seen as divisive and increasingly partisan (Balz, 2016), the developments made public in the years that followed have shaped our understanding of the role of media in future political elections. One of the most notable developments was the Facebook-Cambridge Analytica scandal of 2018. It was revealed that the Trump campaign hired political-consultancy company Cambridge Analytica, to run social media ads, targeting voters in key swing states (Sherr, 2018). The company made use of Facebook ads, private consumer data and media to influence voters' opinions of political candidates (Sherr, 2018). This scandal brought to light the power that social media may have on the outcome of elections. However, it is not just the ability to target key voters but also the type of media that they are shown, that has shaped discussions surrounding political strategy. Specifically, the rise of news with an extremist political outlook. The 2016 presidential election also played a role in these developments. The election saw the increased presence of the likes of Steve Bannon, an aid to Donald Trump and a founder of the right-wing media company, Breitbart news. Although ultimately fired by the Trump campaign, these events have brought to light the potential role that extremist media plays in mainstream politics. These two major developments; the presence of extremist news and the role of social media in distributing this news, has had a notable impact on public opinion and political strategy (Jurkowitz et al., 2021).

The proliferation of extremist news found on social media has had a significant impact socially as well as academically. Companies such as Facebook, Twitter and Instagram have had to invest heavily in detecting and mitigating the large amounts of fake claims often found in extremist news (Facebook, 2017; Crowell, 2017; Instagram, 2019). Academically, the need for identifying extremist news outlets as well as fake news detection in practice has led to a plethora of articles in recent years on the various methods that could be used to detect such media (Conroy et al., 2015). These models are necessary due to the large amounts of articles now being published every day on social media platforms, which human moderators cannot keep up with. With the ability for anyone to create a 'news outlet' on social media, the risk of false, populist news littering the internet has risen. This is further exacerbated when so-called 'verified' accounts post misleading content. A 'verified' account on social media often simply implies that the account is run by who it claims to represent, but this may be deceiving to the public as 'verified' could be interpreted as 'trustworthy' or 'truthful'.

'Verified' accounts distributing fake news may be particularly troublesome in hyper-partisan news outlets. Hyper-partisan, in this paper, is defined as being of extreme partisan ideology or heavily biased of one political outlook. These extreme 'left-wing' or 'right-wing' news outlets cater to a specific audience and may be more inclined to spread deceptive news to further an agenda and leverage off of populist emotions (Silverman et al., 2016).

With increasing pressure for social media companies and regulators to control this rise of deceitful extremist news, there lies significant motivation in being able to detect these deceptive hyper-partisan articles quickly. One way to do so is through linguistic analysis. Linguistic analysis has proven useful

in fields such as author identification or deception detection, as often a person's language cues can be personal or change when they are telling the truth versus a lie (Fuller et al., 2015). Using linguistic analysis, it is of interest to investigate whether there is a statistically significant difference in 'left-wing' and 'right-wing' writing styles and if there is a possibility to accurately identify a text's political ideology. Furthermore, it is of interest whether these linguistic findings can also be used to effectively detect fake news in an effort to mitigate the spread of deceptive articles. These ideas lead to the following research question: ***To what extent do linguistic cues in hyper-partisan news outlets aid in political ideology classification?***
Hence, the primary focus of this paper is to identify linguistic differences in right and left-wing news and to build several classification models using these linguistic differences.

The remainder of this paper will be as follows. Section 2 will explore related works through a literature review. Section 3 discusses the data which will be used and Section 4 the methodology of the paper. Section 5 will reflect the results obtained, Section 6 discusses the limitations and Section 7 will conclude the paper.

2 Literature Review

The rapid rise of partisan news, specifically throughout social media, has led to researchers debating what the root cause may be. To understand the forces behind the spread of deceitful partisan news, Allcott & Gentzkow (2017) argue that one must understand the economics behind fake news. Allcott & Gentzkow (2017) put forward the idea that consumers are presented with a utility trade-off when reading news. On the one hand, readers derive utility from consuming truthful and accurate news. On the other hand, consumers similarly get utility from the news which confirms their views or state of the world. This trade-off is what has allowed hyper-partisan fake news to spread. This is due to the fact that news outlets do not care directly about telling the truth but rather have an economic incentive to generate utility for their readers which then translates into increased advertising revenue due to high readership. News outlets must then choose which form of the aforementioned utilities to appeal to for readers. With the internet and social media, the barrier to entry when creating a 'news outlet' has dropped significantly. This has led to websites being created with the sole intention of creating provoking articles, which generate many readers and hence advertising revenue only to be shut down as quickly as they started. Evidence of this practice was highlighted by Allcott & Gentzkow (2017) who write about investigations which found that 100 fake news websites were run by Macedonian teenagers from the village of Verdes. Allcott & Gentzkow (2017) go on to mention that many of the hyper-partisan fake news websites crucial to the 2016 United States election no longer exist. This evidence showcases the economic model of these partisan websites, where a long-run reputation in truthfulness is exchanged for short-run profits through shocking news (Allcott & Gentzkow, 2017). The authors further argue that this increased level of producers of partisan news within the market leads to a variety of externalities. Among these externalities are the undermining of legitimate news sources and the decrease in the ability for people to make informed decisions when voting. Perhaps the most interesting idea put forward by

Allcott & Gentzkow (2017) is the self-reinforcing cycle that this fake partisan news creates. As fake news also reduces the trust in legitimate sources, 'credible' news may have to resort to participating in the utility trade-off. This means trading in the news which appeal to the utility derived from reading the truth and invest in producing partisan which caters to the other utility source. This skewed incentive scheme further highlights the utility trade-off faced by both suppliers and consumers. This phenomenon, which became a central discussion after the 2016 United States presidential election, has bolstered the research into analysing and identifying deceitful, hyper-partisan news. One such method is linguistic analysis.

As statistical and computational techniques have developed, so has the literature surrounding linguistic analysis models. Conroy et al. (2015) explore not only the use of a psychological linguistic approach in analysing texts, as followed in this paper, but additionally put forward a network approach. A network approach uses a pre-existing knowledge base with which it compares the content of the article being analysed. It uses aspects such as the meta-data of the content to create a network path to the pre-existing knowledge base. Here, the shorter the path, the more likely it is that the given article is similar to the original knowledge base (Conroy et al., 2015). Linguistic analysis has many uses, one primary one being deception detection. The reason linguistic analysis is useful in deception detection lies in the idea that cognitive processes change when lying (Fuller et al., 2015). Giolla et al. (2017) address this concept in their review of literature arguing for the usefulness of using drawings in deception detection. As with Fuller et al. (2015), Giolla et al. (2017) argue that people who lie experience a stronger cognitive load, due to the need to create details that never actually occurred. Furthermore, liars often prepare their story verbally, but when asked to relay their story using a different medium, such as drawing, the lack of preparation and ability to conjure up visual details, may make it easier to tell liar apart from those telling the truth (Giolla et al., 2017). This idea, that a lack of cognitive flexibility may allow for the detection of liars, was additionally argued by Hauch et al. (2015) who conducted a meta-analysis of 44 studies concerning linguistic cues in deception detection. They found that liars not only experienced greater cognitive load but also used more negative emotions, use fewer words related to perceptual or sensory experiences, distanced themselves and try to refer to cognitive processes less often compared to truth-tellers. These findings, that language use changes according to the cognitive state an author may be in, could be expanded to other forms of textual classification. If writers change their style while lying due to psychological effects, could the different writing styles of left and right wing authors be detected and classified? It may be that left and right wing authors experience different cognitive loads, or are in different psychological states and hence develop different writing patterns. This is the premise of this paper's decision to use linguistic cues and language models to analyze hyper-partisan news and investigate whether language usage can indicate the political orientation of an author.

Hyper-partisan news presents an opportunity to apply language analysis tools in a practical manner and whose results may have significant insights into political strategy. Potthast et al. (2017) analyzed a data set created by BuzzFeed to try to determine stylistic differences in left-wing and right wing news outlets, as is the central focus of this paper. Using a method known as Unmasking, the authors found that there was a larger stylistic similarity between the two than either had with 'mainstream' news outlets.

Unmasking is a process put forward by Koppel et al. (2007), and was originally developed to determine whether two texts are written by the same author or not. Koppel et al. (2007) had the idea that texts from a singular author would have very few differentiable features between them. Comparatively, texts written by different authors would contain a larger amount of differentiable features. This is the main premise of Unmasking. First, Koppel et al. (2007) compile a list of features that are different for the two texts being analysed. Each text is then broken into chunks of roughly 500 words. Once compiled, an iterative process is constructed to build what is called a degradation curve. In essence, in each step, k of the most important differentiating features between the texts are removed. The authors then predict which chunk belongs to which text using the remaining differentiable features. The accuracy is reported and is plotted for each iteration. In theory, since works by the same author contain fewer differentiable features, these texts will become harder to predict at a much faster rate than those texts written by different authors. This is reflected in the degradation curves. This process is repeated until a graph such as Figure 1 is achieved.

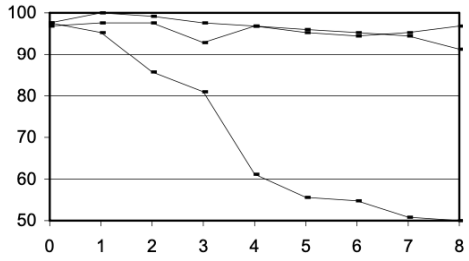

Figure 1: Example of degradation curves (Koppel et al., 2007).

Figure 1 shows the degradation curves for multiple texts which are compared to a single text, the benchmark. The lines which remain at the top are written by different authors whereas the line steadily decreasing relates to the text written by the same author as the benchmark text. These curves allow the reader to visualize the process of Unmasking and using these curves, determine which texts are likely written by the same author.

Where unmasking is one method of determining the author of a text, a similar method of analysis is conducted in this paper. However, the text-mining software Linguistic Inquiry and Word Count (LIWC), which will be discussed more in-depth in Section 4, will be used.

LIWC is a popular tool in language research. Cutler et al. (2020) employ LIWC as well as a machine learning model to predict whether a person suffers from narcissism based on their language usage. Similarly, van der Zee et al. (in press) run LIWC on a collection of tweets by former United States President Donald Trump. The results allow for, among other things, the prediction of the veracity of Trump's tweets. Finally, Reis et al. (2019) turn to, in conjunction with other features such as article engagement or lexical features, LIWC to develop a deception detection model. In terms of predicting, van der Zee et al. (in press) construct a logistic regression (logit) model and compare its predictive ability to a wide array of other models found in similar research. The authors find that their logit model outperforms all other cases in terms of accuracy. Reis et al. (2019) look at the collection of aforementioned features to predict fake news using multiple classifier models. It is found that the model, eXtreme Gradient Boosting

(XGBoost), outperforms almost all other classifiers (Reis et al., 2019). Given the success found in the two previously mentioned papers, this paper seeks to run LIWC on the hyper-partisan data-set based on the research of Potthast et al. (2017), use the approach proposed by van der Zee et al. (in press) for political ideology classification rather than deception detection, and finally, extend upon these models by running the XGBoost classifier utilized in Reis et al. (2019).

Based on this outline, the following hypotheses are put forward.

H1: True and false articles will have statistically significant linguistic differences. Although not the central focus of the paper, first investigating differences in true and false articles in both left and right wing publications, may provide additional insights into using language for classification.

H2: When comparing over all articles, right-wing and left-wing texts will have statistically different language usage. One thing to be noted is that LIWC makes use of language variables such as comma usage but also analyses topical variables, such as money and religion. To avoid using variables that may be inherently left-wing or right-wing, all topical variables, as identified by van der Zee et al. (in press), will be excluded from the analysis.

H3: Political ideology classification methods using linguistic variables will outperform random guessing models

3 Data

A data set initially constructed by BuzzFeed will be used to analyse hyper-partisan news. This data set was created by Silverman et al. (2016). A team of data journalists analysed nine news outlets and their posts spanning 19-23 September 2016 as well as 26 and 27 September 2016. The posts were of a political nature and often surrounded the 2016 presidential election. Three of these outlets were right-wing, three left-wing and three were mainstream outlets. All of the chosen news outlets had a Facebook verification mark, denoted by a blue checkmark next to the name. A team of fact-checkers checked a total of 2282 posts, of which 1145 were mainstream, 666 right-wing and 471 left-wing. One fact-checker would review a post and if unsure this post would be checked by a second fact-checker. A third fact-checker was brought in if there was a disagreement between the first two. The third fact checker also reviewed all articles labelled 'mostly false'. There were four rankings available for each article; mostly true, mixture of true and false, mostly false and no content. The labels, 'mostly true' and 'mostly false' indicate that the articles were deemed close to entirely true or false respectively, with slight variation allowed. The label 'mixture of true and false' implies that the article contains both elements that are factually incorrect as well as correct. Due to the article containing factually incorrect information, the categories 'mostly false' and 'mixture of true and false' will be grouped as false, as done in previous literature (Potthast et al., 2017; Reis et al., 2019). The label 'no content' refers to when the outlets would post images or opinions with no factual claims. This last category is not used for analysis in this paper.

Accessing this data is troublesome, however, as the data was not directly provided, only website links to the initial posts and many of these web articles have since been removed or expired. Due to this

difficulty, Potthast et al. (2017) conducted an analysis on the aforementioned data set. The authors combed through the BuzzFeed data set and created an archive of the actual article content which could be extracted from XML files[1]. Potthast et al. (2017) were unable to extract all the articles and could only archive 1627 articles, of which 545 were right-wing, 826 mainstream and 256 left-wing. Data cleaning was conducted on this archived data set provided by Potthast et al. (2017). All entries that were empty due to no longer containing text were removed, resulting in 38 articles being dropped. Furthermore, the 61 entries labeled as 'no factual content' were removed and the categories of mostly false and 'mixture of true and false' were combined. This resulted in a final data set of 1529 entries; 475 right-wing, 810 mainstream and 244 left-wing. The breakdown of the data into true and false categories can be found in Table 1 below. Mainstream data is included for contextualization as well as to provide additional insight into the sentiment analysis discussed later in this paper.

Table 1: Final cleaned data set showing number of articles labelled true or false

	Right-wing	Left-wing	Mainstream	Total
True	262	178	802	1242
False	213	66	8	287
Total	475	244	810	1529

Note: The variable label 'True' corresponds to the dataset label 'mostly true', similarly 'False' corresponds to the the sum of articles labelled 'mostly false' and 'mixture of true and false'

4 Methodology

This paper primarily seeks to replicate and extend upon the work of van der Zee et al. (in press) but with a focus on hyper-partisan analysis instead of deception detection. Firstly, using the cleaned data set discussed in Section 3, LIWC will be run on all articles. LIWC is a text-analytics program catered towards classifying words into psychologically relevant categories. Essentially, the software looks at each word in a text and determines whether this word is present in any of the pre-defined psychological dictionaries. The end results are the percentage of words in the text which belong to a variety of psychological categories.

Once the values of LIWC have been obtained, one additional variable will be constructed; a sentiment score using the Valence Aware Dictionary for Sentiment Reasoning (VADER) [2]. VADER is a sentiment analysis tool that uses a dictionary to analyze text and returns a sentiment score between -1 and 1. Due to its use of a dictionary, it is often more computationally efficient than machine learning methods (Hutto & Gilbert, 2014). Besides computational efficiency, one of the main advantages of VADER is that it has been specifically constructed to analyse social media communications and blogs (Hutto & Gilbert, 2014). This is due to the fact that the dictionary includes popular internet jargon and slang as well as emoticons. The premise of the tool is that each word within the dictionary is assigned an integer value between -4 and 4 by 10 independent raters. A value of -4 indicates that the word is of extremely negative sentiment whereas the opposite holds for words with a value of 4. The 10 scores of each word, given by the 10 independent raters, are then averaged. This average is the final sentiment score for each word or emoticon in the constructed dictionary. Finally, the sentiment scores of each word in the article are

[1] Obtainable at: https://zenodo.org/record/1239675.YJbAGS0RrVp
[2] More on VADER can be found here: https://github.com/cjhutto/vaderSentiment

summed and normalized in order to arrive at a final sentiment score for the entire article between -1 and 1. This normalization technique is shown below in equation 1.

$$s_i = \frac{\sum_{j=1}^{J} k_{i,j}}{\sqrt{\left(\sum_{j=1}^{J} k_{i,j}\right)^2 + \alpha}} \tag{1}$$

Here s_i is the compound sentiment score of article i. Further, $k_{i,j}$ is the sentiment score for word j in article i, given by the constructed dictionary. Looking at the source code, the α in the denominator is set to fifteen. This is a deliberative choice made by the creators of the VADER package. It is meant to approximate the expected maximum value of the numerator and its value is static, meaning it is not supposed to be changed. Through this normalization equation, article i arrives at a single compound sentiment score between -1 and 1.

In order to compare linguistic cues and the language use of left wing and right wing outlets, the same Multivariate Analysis of Variance (MANOVA) model will be used as in van der Zee et al. (in press). The MANOVA will allow for a comparison of the means of all LIWC and sentiment variables in true and false articles where the left-wing and right-wing news pieces are combined. Similarly, a MANOVA will be used to compare the means of LIWC and sentiment variables in right-wing and left-wing outlets. This analysis will allow for further insights into hypotheses H1 and H2 respectively.

To test H3, the methods of van der Zee et al. (in press), Reis et al. (2019) and Decision Tree Classifiers are used in tandem. A logit model as used by van der Zee et al. (in press), is constructed to predict whether a text is left-wing or right-wing. Hence, the dependant variable is the political orientation of the article, where 1 is right-wing and 0 is left-wing. The independent variables are the LIWC and sentiment variables that are found to be significantly different in right and left articles using the aforementioned MANOVA. As done by the authors, several model construction techniques are analysed to determine which variables to include; forward, backward, using variables significant at 5% and 1% levels respectively and finally, the Least Absolute Shrinkage and Selection Operator (LASSO) method. Here, several metrics will be used to select the model. A comparison of how parsimonious the model is, as well as the Log-Likelihood, Akaike Information Criterion (AIC) and the Area Under Curve is made to determine the best model. The chosen model will provide the variables for the logit model. As mentioned in Section 1, topical variables will not be included in the analysis.

Once the logit is constructed, the same performance metrics will be analysed as with van der Zee et al. (in press). This entails out-of-sample testing. To do this, the data will be split into 70% training and 30% testing sets. Using the out-of-sample predictions, this paper seeks to investigate the Receiver Operating Characteristic (ROC) curve as well as the Area Under the Curve (AUC). A ROC curve has the False Alarm rate, or false positive rate, on the x-axis and Hit rate, or true positive rate, on the y axis. A false positive is when the model incorrectly classifies a prediction as positive while a true positive is when it correctly classifies a prediction as positive. The curve illustrates the varying Hit and False alarm rates offered by a model when different cut-offs are used for classification. An example of a ROC curve has been added below, taken from an article by Brownlee (2018).

In Figure 2, random guessing would be the diagonal line which has an AUC 0.5. A ROC curve above

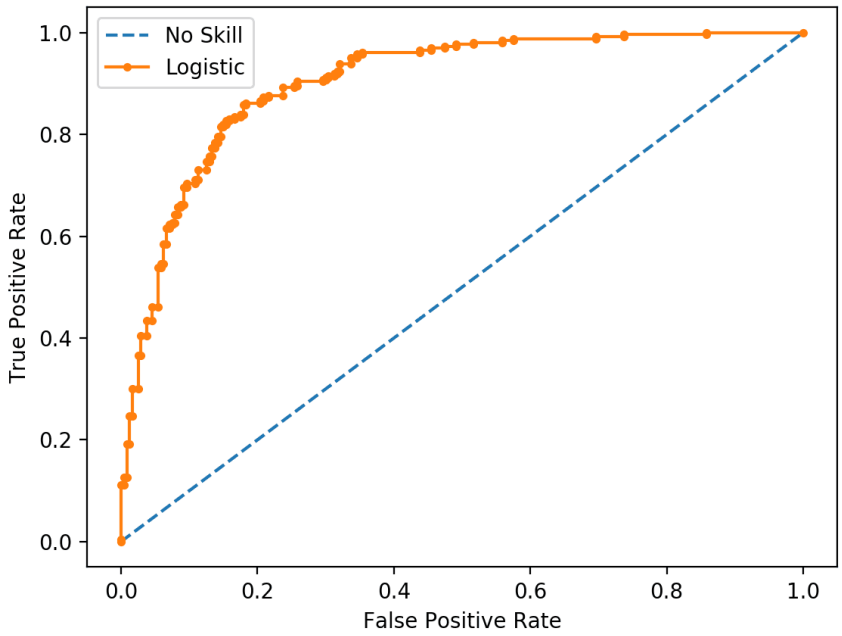

Figure 2: Example ROC curve for a logit model as well as random guessing (Brownlee, 2018).

the ROC concerning random guessing indicates an improvement in predictive ability (van der Zee et al., in press).

Additionally, two models will be added for extension. Reis et al. (2019) look at a variety of classification techniques and their predictive ability. One particular model, known as XGBoost, is found to offer promising results.

XGBoost is a classification technique that utilizes tree structures and machine learning algorithms and whose success has seen it be widely adopted as an efficient and powerful classification technique. This paper will run XGBoost, with the same variables that will be in the logit model.

XGBoost's advantage is its implementation of a process known as boosting. Boosting uses a series of weaker classifiers to build a much better performing classifier. This is done through its creation of multiple trees, unlike other Decision Tree methods which only employ one tree. To illustrate this, Figure 3, which is taken from an article by Hoffstein (2020), shows a simple example of boosting.

Using the example in Figure 3 below, it can be seen how boosting operates. In this example, the model seeks to classify the '+' and '-' within the box using various rules or splits, here the dotted lines. The first classifier is constructed on the original train data set, with the split being the top horizontal line. The split puts forward the rule that all the values above the line are a '-' whereas those under are a '+'. Here the miss-classified observations are circled. XGBoost then updates the weights, giving a smaller weight to correctly specified values and a larger weight to those it miss-classified. This can be seen in the first 'Updated Weights in dataset' box. XGBoost then tries a new split, the vertical line. All values to the left are classified as '-' while those on the right of the split are '+'. It checks for any miss-specifications and once again updates the weights. This process repeats and is known as boosting. Once XGBoost has

created these three weak classifiers, it combines them to create a much better performing classifier, based on what it has learned from the previous three.

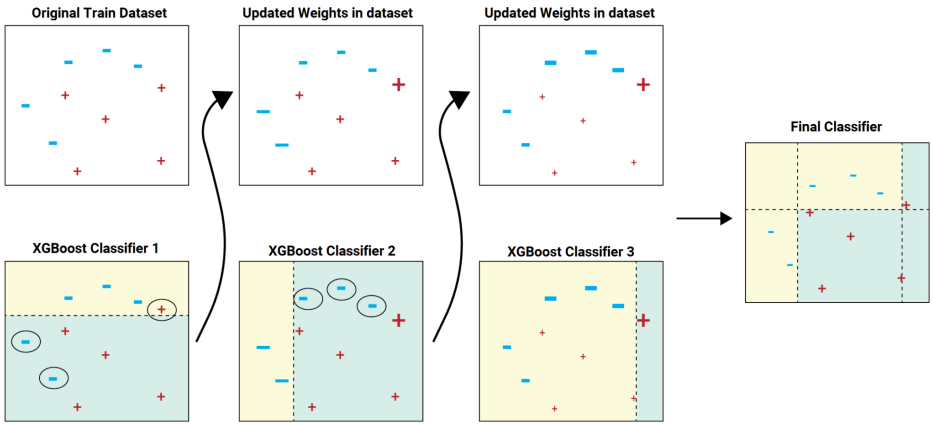

Figure 3: Example of how boosting operates (Hoffstein, 2020).

Another tree classification model will be used in order to assess the relative performance of XGBoost to similar tree classification methods. The Decision Tree Regression (DTR) only makes use of one tree, compared to XGBoost which uses multiple. Using non-parametric decision rules, a tree structure is created which minimizes an impurity value at each node. In this paper, the Mean Squared Error (MSE) is the chosen impurity value. A node with N observations is split as follows. DTR selects an explanatory variable, in this paper the chosen LIWC or sentiment variables. The chosen variable is denoted as x_i, where i represents the value of the variable for observation i. It calculates the sample average of the chosen variable within the node, as seen in equation 2,

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{2}$$

.

Once this sample average is calculated, the DTR splits the sample according to the chosen variable and a threshold. This split results in two sub-samples with the observations either being below or above the threshold according to their value of x_i. Let these sub-samples be denoted by $k = 1,2$. Once these sub-samples have been created, the DTR calculates the MSE for each sub-sample, k, according to equation 3,

$$MSE_k = \frac{\sum_{i=1}^{N_k} (x_{k,i} - \bar{x}_k)^2}{N_k} \tag{3}$$

Finally the impurity value (IV) for this variable-threshold choice can be calculated. This is the weighted MSE of each sub-set shown in equation 4,

$$IV = \frac{N_1}{N} * MSE_1 + \frac{N_2}{N} * MSE_2 \tag{4}$$

.

DTR iteratively goes through the thresholds for a specific variable and keeps the lowest IV value. This entire process is repeated over all variables. The variable-threshold combination which leads to the lowest IV over all variables is used to split the node. All nodes within the tree are split according to this process until the maximum depth of the tree or the minimum split value within the node is reached.

DTRs can easily be visualized due to their structure and the above-mentioned generated rules give insights into the role of the variables in the model, unlike many 'black box' machine learning models.

A key part of DTRs and other tree models, is the choice of hyper-paramters, such as maximum depth and minimum split mentioned earlier. These hyper-paramters often play a significant role in minimizing overfitting (Lewis, 2000). The concept of hyperparameter tuning is known as pruning and a key aspect of effective model selection. Maximum depth represents the maximum number of levels within the tree. Minimum split is the minimum amount of observations needed in a node for it to be further split into two new nodes. Mantovani et al. (2018) explore the effects of hyperparameter choices in decision trees and find that decision trees start performing well from a maximum depth of 4 onwards. The authors find that the ideal maximum depth is not entirely clear-cut as performance is fairly uniform between 4 and 30. They put forward that this indicates that the ideal choice may be more based on aspects such as the number of observations. In terms of the minimum split, Mantovani et al. (2018) report that the best performance is achieved with a minimum split value between five and ten. In this paper, at first, a maximum depth of 6 and a minimum split of 10 will be used for the DTR. For XGBoost, a maximum depth of 7 will be used. It must be noted, however, that for robustness, the hyper-parameters will be tuned to determine if better performance can be achieved. This hyper-parameter tuning will be further discussed in Section 5.

Lastly, the final hyper-parameter choice for XGBoost must be made, which is the number of trees that will be constructed for boosting. Here a choice of 100 and 1000 will be used and the relative performance will be assessed to see if a substantial difference arises.

In summary, this paper seeks to investigate the linguistic differences in left-wing and right-wing news and its ability to aid in political ideology detection. Using LIWC and sentiment variables, the differences between hyper-partisan news are explored using a MANOVA as well as differences in true and false articles. A logit model will be constructed as done by van der Zee et al. (in press) to predict the political orientation of texts. As an extension, two new models, XGBoost and DTR will be constructed to determine if an improvement can be made over the performance of the logit model.

5 Results

5.1 Sentiment Analysis

Using the VADER sentiment package, all articles are given a sentiment score between -1 and 1. As covered in Section 4, a score of -1 indicates an extremely negative sentiment whereas 1 indicates extremely positive sentiment.

Table 2 and 3 display the spread of the sentiment within left-wing and right-wing articles respectively. The sentiments are grouped into intervals of 0.2. As an example, in the first row, an upper limit of -0.8

Table 2: Left wing frequency data

Upper limit	Frequency	Percentage of total
-0.8	141	57.79%
-0.6	15	6.15%
-0.4	4	1.64%
-0.2	5	2.05%
0	1	0.41%
0.2	2	0.82%
0.4	4	1.64%
0.6	3	1.23%
0.8	5	2.05%
1	64	26.23%

Table 3: Right wing frequency data

Upper limit	Frequency	Percentage of total
-0.8	243	51.16%
-0.6	24	5.05%
-0.4	10	2.11%
-0.2	9	1.89%
0	8	1.68%
0.2	5	1.05%
0.4	7	1.47%
0.6	13	2.74%
0.8	15	3.16%
1	141	29.68%

indicates that these are articles with a sentiment between -1 and -0.8.

As the tables show, around 57.79% of left-wing articles have a sentiment between -1 and -0.8, whereas this applies to 51.16% of right-wing articles. On the other side of the spectrum, only 26.23% of left-wing articles are between 0.8 and 1 and 29.68% of right-wing articles are between 0.8 and 1. This illustrates that both left-wing and right-wing articles tend to be biased towards negative sentiment, with very few neutral articles being found within the dataset for both political orientations. Furthermore, 68.03% of left-wing articles are between -1 and 0, and 61.89% of right-wing articles are between -1 and 0. This negative bias within hyper-partisan news is made more apparent when compared to the sentiment distribution of mainstream news.

Table 4: Mainstream Sentiment frequency data

Upper limit	Frequency	Percentage of total
-0.8	282	34.81%
-0.6	34	4.20%
-0.4	16	1.98%
-0.2	13	1.60%
0	14	1.73%
0.2	9	1.11%
0.4	21	2.59%
0.6	21	2.59%
0.8	26	3.21%
1	374	46.17%

Table 4 shows that mainstream news tends to be more positive than hyper-partisan articles. With just 34.81% of mainstream articles having a sentiment score between -1 and -0.8, this is substantially different from both left and right wing articles. Whereas left and right wing articles did not have more than 30% of their articles in the 0.8 to 1 sentiment range, 46.17% of mainstream articles were within this range. Finally, only 44.32% of mainstream articles had a sentiment score between -1 and 0. This stark difference between mainstream and hyper-partisan news points towards the potentially different writing styles of the two and how hyper-partisan news may try to capitalise off of negative emotions.

Looking at the cumulative distribution of sentiment in Figure 4, this relationship is made clear. Here we see the initial dominance of negative sentiment in both left and right wing articles. It seems that left-wing sentiment has a first-order stochastic dominance over right-wing news. However, both seem to

converge at the tail ends of the graph, making it unclear if the stochastic dominance is definitive. What the graph does show, is the fact that left wing news has slightly more negative sentiment, expressed by the higher cumulative frequency line. Right wing news has slightly more positive articles, evidenced by the steeper cumulative frequency line towards the right-hand side of the graph. Despite these differences, both have first-order stochastic dominance over mainstream news. Further, there is no apparent second-order stochastic dominance as none of the lines cross. This is indicative of a similar variance among all sentiment distributions. Additionally, what Figure 4 shows is the lack of neutral articles within all three news sources. All lines start to trend mostly sideways as the graph progresses along the x-axis, showing a steady but slower increase in cumulative frequency and hence a lack of sentiment-neutral articles. This trend changes, however, as it approaches the more extreme positive sentiment. Here one can see how mainstream news contains more positive articles, as the cumulative frequency line sharply increases, from a much lower position than the left and right wing cumulative frequency lines.

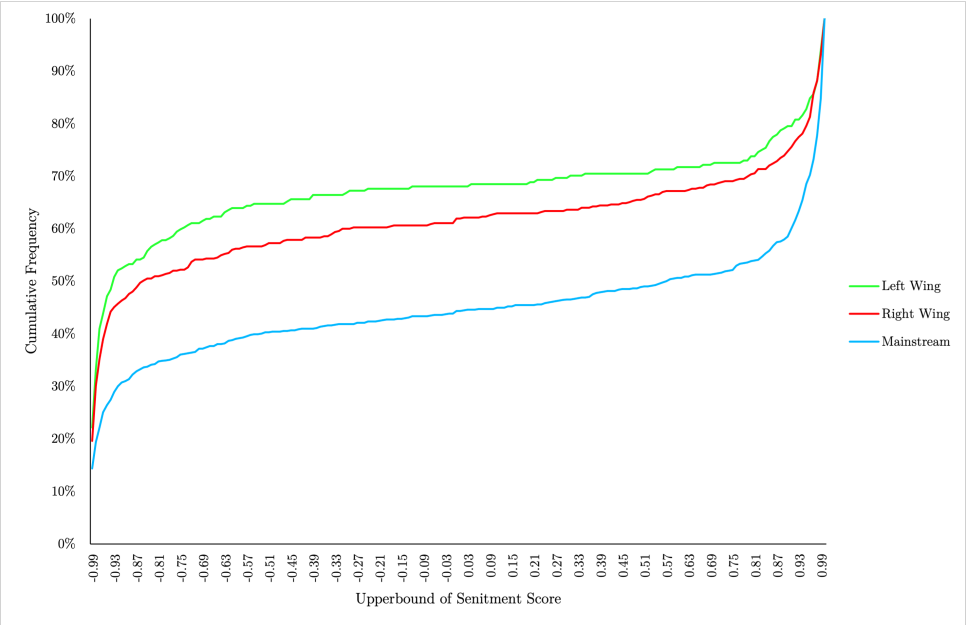

Figure 4: Cumulative Sentiment Distribution

These findings are indicative of a potential difference in writing styles among hyper-partisan and mainstream news. With a bias towards negative sentiment, it strengthens the question whether other linguistic differences exist in hyper-partisan news which may aid in political orientation classification and prediction.

5.2 Analysing Veracity

In order to test H1, a MANOVA is run to determine if there are any statistically significant language and sentiment differences between true and false articles in our data set. Here we combine the left and right wing articles into one sample, and split the data according to veracity, with true articles having a veracity score of 0 and false a value of 1. The resulting statistically significant variables are displayed below in Table 5. The Table uses the original LIWC names as variable names due to space limitations.

In the Appendix, Table 19 provides an explanation as to what each LIWC name represents for ease of interpretation.

Table 5: Significant Variables from MANOVA comparing True and False

Variable_name	Mean_if_true	Mean_if_false	F_stat.	p-value	Sig.	Bayes Factor	Cohen's d	CI
hearing	1.02	0.76	14.08	0.014	*	87.330	0.34	[0.16,0.52]
they	0.93	1.24	13.22	0.014	*	57.970	-0.33	[-0.51,-0.15]

Note. * p <0.05,,** p <0.01, *** p <0.001

As Table 5 shows, only 2 variables are found to be significantly different between true and false articles. Both these variables are significant at the 0.05 level with the p values adjusted using the False Detection Rate (FDR) correction. The FDR correction makes estimates more conservative by correcting for random events that may appear statistically significant (Rouam, 2013; van der Zee et al., in press). Interestingly, there seems to be almost no difference between true and false articles. Given that 82 variables were utilized in this analysis, it is surprising that there seem to be so few differences between true and false articles. This may be, however, due to the conservative choice of using FDR corrections. However, in the works of van der Zee et al. (in press), the same approach was used, also accounting for FDR corrections, and they found many more statistically significant differences in their MANOVA comparing LIWC variables in true and false articles. These findings contrast, not only the findings of van der Zee et al. (in press) but also the large volume of literature previously discussed which found ample evidence of linguistic differences between articles of varying veracity. However, it must be noted that the dataset used in this paper is different from that of other literature. This difference may be the cause of the unexpected results found here and will be further touched upon in Section 7.

5.3 Analysing Political Orientation

Turning to hypothesis 2, a MANOVA is run comparing the mean of 81 LIWC variables as well as the mean of the sentiment variable between left and right wing articles. As mentioned in the Methodology, topical variables are excluded from this analysis. These topical variables are defined by van der Zee et al. (in press). More specifically, these are the categories; Achievement, Biology, Body, Death, Health, Home, Ingestion, Leisure, Money, Religion, Sexual and Work. Unlike the MANOVA for veracity, this MANOVA showed significantly more differences between political orientations. Looking at Pillai's trace, it is noted that the binary variable, political orientation, is statistically significantly different between left and right-wing texts. The test resulted in a Pillai's trace of 0.414, an F-score of 3.627 with (82, 420) degrees of freedom. The effect had a p-value $< 2.2 * 10^{-16}$. In total, 25 variables were found to be significantly different between left and right wing articles. Of these, 7 were statistically significant at the 0.001 level, 9 at the 0.01 level, and the remaining 9 at the 0.05 level. The statistically significant different variables are reported in Table 6 below with the mean of each variable in left and right wing texts. Further, the p values are adjusted with the FDR correction. Additionally, the Bayes factor from a Bayes t-test is reported as well as Cohen's d and the confidence interval of said Cohen's d. The full results of the MANOVA are reported in Tables 16 - 18 in the Appendix.

Table 6 makes a few interesting results apparent. Left wing articles contain, on average, a higher pro-

Table 6: Significant Variables from Manova Left vs Right

Variable	Mean_if_Left	Mean_if_Right	F_stat.	p-value	Sig.	Bayes Factor	Cohen's d	CI
Apostro	1.51	2.18	28.75	0.000	***	>1,000	-0.50	[-0.69,-0.32]
Colon	0.46	0.35	6.50	0.038	*	2.390	0.24	[0.05,0.43]
Comma	4.91	4.34	14.82	0.001	**	126.500	0.36	[0.18,0.55]
conj	5.19	5.55	7.54	0.026	*	3.940	-0.26	[-0.44,-0.07]
Dash	0.85	0.58	18.95	0.000	***	885.120	0.41	[0.22,0.60]
dic	78.18	80.92	26.19	0.000	***	>1,000	-0.48	[-0.67,-0.29]
exclam	0.09	0.25	10.50	0.008	**	16.200	-0.30	[-0.49,-0.12]
female	0.49	1.28	26.74	0.000	***	>1,000	-0.49	[-0.67,-0.30]
focuspresent	7.81	8.51	6.99	0.031	*	3.030	-0.25	[-0.43,-0.06]
friend	0.12	0.21	10.18	0.008	**	13.940	-0.30	[-0.49,-0.11]
function	46.92	48.67	11.75	0.005	**	29.460	-0.32	[-0.51,-0.14]
I	0.61	0.87	7.25	0.029	*	3.440	-0.25	[-0.44,-0.07]
male	2.54	1.77	20.94	0.000	***	>1,000	0.43	[0.24,0.62]
motion	1.36	1.62	11.99	0.005	**	32.980	-0.33	[-0.51,-0.14]
periods	4.91	5.62	27.82	0.000	***	>1,000	-0.50	[-0.68,-0.31]
ppron	5.26	6.05	10.03	0.008	**	12.980	-0.30	[-0.48,-0.11]
pronouns	10.43	11.42	9.17	0.012	*	8.600	-0.28	[-0.47,-0.10]
QMark	0.26	0.39	6.93	0.031	*	2.950	-0.25	[-0.43,-0.06]
Quote	1.48	2.05	11.50	0.005	**	26.190	-0.32	[-0.51,-0.13]
Sixltr	23.46	22.38	6.29	0.041	*	2.160	0.24	[0.05,0.42]
social	9.99	10.91	8.55	0.016	*	6.400	-0.28	[-0.46,-0.09]
they	0.83	1.16	13.79	0.002	**	77.530	-0.35	[-0.54,-0.16]
verb	13.71	14.59	10.12	0.008	**	13.510	-0.30	[-0.49,-0.11]
WPS	22.69	19.39	35.86	0.000	***	>1,000	0.56	[0.38,0.75]
you	0.56	0.78	8.84	0.014	*	7.360	-0.28	[-0.47,-0.09]

Note. * p <0.05, ** p <0.01, *** p <0.001

portion of male references (*male*) whereas right-wing articles contain, on average, a higher proportion of female references ($female$). Left wing articles have longer sentences (WPS) yet contain fewer dictionary words (*dic*). Right-wing articles tend to use more first-person personal pronouns (I), as well as make use of more quotation marks (*Quote*). This lends itself to the idea that right-wing news possibly makes more use of quotes within their articles as well as focusing on more reader-centric language, supported by their use of first-person pronouns (I). The MANOVA also indicates that right-wing articles have a higher proportion of second and third-person pronouns (through *you* and *they*) as well as a higher proportion of total personal and other pronouns (indicated by *ppron* and *pronoun* respectively). This may be an indication that right-wing texts use more grouping words such as 'they' or 'them' in an effort to distinguish between certain parties within their texts. Lastly, in both the MANOVA comparing true and false as well as the MANOVA comparing left and right, the difference in the average of the sentiment variable was unfortunately found to be statistically insignificant. This is supported by the sentiment analysis described earlier, where both political orientations shared a similar negative sentiment bias.

Now that the statistically significant variables have been identified, a choice must be made as to which of the variables to use in the logit model.

As in the paper by van der Zee et al. (in press) as well as mentioned in the Methodology of this paper, five variable selection techniques will be used. The results of said variable selection techniques can be found in Table 7 below.

As can be seen in Table 7, all five constructed logit models, regardless of variable choice, have an AUC above 0.83, an already promising result. Using all variables significant at at least the 5% level delivers the highest AUC of 0.847. In terms of AIC, the Backward variable selection technique offers the lowest value of 495.84, closely followed by 497.40 using the Forward selection technique. This is to be somewhat

Table 7: Results of varying variable selection techniques

Model	Forward	Backward	Sig. 5%	Sig. 1%	LASSO
Number of variables	15	13	25	16	20
Log-likelihood	-232.70	-233.92	-229.51	-235.57	-229.97
AIC	497.40	495.84	511.01	505.14	501.94
AUC	0.844	0.842	0.847	0.839	0.846

expected for both of these techniques, as lowering the AIC is the main criteria for these two variable selection techniques.

For this paper, the Forward model is chosen. This is due to a balance of parsimony as well as a high AUC and low AIC. The Forward model is more parsimonious than all but the Backward model. The choice between the Forward and Backward models seems to be a matter of preference as the difference in AUC and AIC is not that high. The Forward model has slightly more variables than the Backward. It is of interest to investigate the full extent language variables can have on the prediction of political orientation. Hence this paper moves forward with the slightly less parsimonious Forward model, in favour of having more variables to analyse.

Now with the variables selected, a logit model is run on the training set. Firstly, the marginal effects of the utilized variables are displayed below in Table 8.

Table 8: Marginal effects of logit model variables using variables from the Forward selection technique

Variable Name	Marginal Effect	Standard Error	z score	p value
WPS	.005	.006	.848	.396
male	-.075	.013	-5.573	0.000
Dic	.022	.005	4.376	0.000
Apostro	.056	.020	2.850	.004
female	.067	.021	3.252	.001
Period	.082	.025	3.265	.001
Exclam	.233	.083	2.813	.005
Comma	-.037	.014	-2.670	.008
focuspresent	-.033	.011	-3.061	.002
motion	.063	.030	2.094	.036
they	.068	.029	2.325	.020
Quote	.033	.015	2.176	.030
Dash	-.072	.032	-2.225	.026
friend	.177	.097	1.824	.068
QMark	.081	.055	1.473	.141

With right wing articles having a value of 1 and left wing a value of 0, we can interpret the effects of each chosen variable using Table 8. As expected, and mentioned in the analysis of the MANOVA results, we once again see variables such as male and female playing a substantial role between left and right wing texts. Specifically, a one percentage point increase in the variable female leads to a 6.7 percentage point increase in the probability of a text being right-wing. Inversely, a one percentage point increase in the variable male leads to an even larger 7.5 percentage point increase in the probability of an article being left-wing. Interestingly, the variables which have the largest absolute marginal effect are the variables 'Exclam' and 'friend'. Referring to Table 19 in the Appendix, we see that 'Exlam' refers to exclamation marks and 'friend' to Friends. These results further support the notion that right-wing texts

make use of more emotive language as well as grouping language, through the use of exclamation marks and friend-oriented language respectively. A one percentage point increase in exclamation marks leads to a substantial 23.3 percentage point increase in the probability that a text is right-wing. Similarly, a one percentage point increase in the variable friend, leads to a 17.7 percentage point increase in the probability that a text is right-wing. These results indicate that there is a significant role to be played by language variables in classifying the political orientations of articles. Furthermore, a few of these specifically chosen variables show promising and large marginal effects which may result in strong predictive performance for the models described in the following sections.

5.4 Logit Results

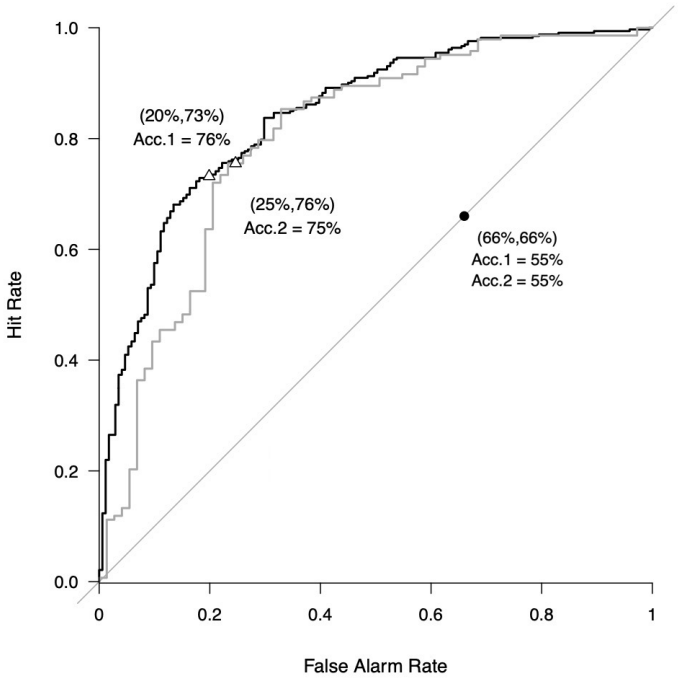

Figure 5: ROC curve for the logit model

Once the marginal effects of the logit model have been investigated, we move to an analysis of said logit model's performance both on the training data set as well as the test data set. Figure 5 displays the ROC curve of the logit model. The black line is the ROC curve of the training data and the grey the ROC curve of the test data set. In this paper, a hit (referring to the Hit Rate in the ROC graph) is when a prediction is correctly classified as right-wing. A false alarm is when a prediction is classified as right-wing when it is in fact left-wing. As can be seen, and is expected, the model performs better on the training than the test data set. Nevertheless, the given accuracy of both is very close, with an accuracy of 76% on the training set and 75% on the test set. The coordinates correspond to the hit rate and false alarm rate (FAR) respectively when a cut-off of 66% is used. The straight line represents the ROC of random guessing using various cut-offs. The given point relates to random guessing with a cut-off point of 66%. As can be seen this results in an accuracy of 55%, substantially lower than the logit model for

both training and test sets.

Taking a further look into the performance of the logit model, Table 9 below provides additional insights. Specifically, the performance of the logit not only using as a cut-off the mean of the training data set (66%) but also using a cut-off of 50%. Data set 1 refers to the training data set and data set 2 the test data set.

What this table shows is how close the performance of the logit model is when analysing the test and training sets. Looking at the cut-off of 50% we see an accuracy of 78.13% in data set 1 and 78.24% in the test data set. These accuracies seem to be close with fairly similar confidence intervals, although the confidence interval of the test set is understandably slightly wider. Even though these accuracies are above the performance of the logit model using a 66% cut-off, there is a large pitfall when the hit rates and false alarm rates (FAR) are investigated. While the hit rates seem to be substantially higher with a cut-off of 50%, so are the false alarm rates. In data set 1, the false alarm rate is almost double the false alarm rate with a cut-off of 66%. While the FAR increases for the 66% cut-off in data set 2, there is still a large difference between the two cut-offs. Although a higher hit rate is desirable, it remains questionable if the trade-off of a much higher false alarm rate is acceptable.

Table 9: Performance of Logit model on training (1) and test (2) dataset

Dataset	Cut-off	Accuracy	Acc. CI	Hit rate	FAR	Precision	F1	AUC	AUC CI
1	50.00	78.13	[74.55, 81.31]	87.65	40.35	80.83	84.10	84.40	[80.80, 88.00]
	66.00	75.55	[71.17, 79.72]	73.19	19.88	87.73	79.80		
2	50.00	78.24	[71.76, 82.42]	86.71	38.36	81.58	84.07	80.10	[73.50, 86.70]
	66.00	75.46	[57.87, 83.33]	75.52	24.66	85.71	80.30		

Note. All values are in percentages (%). F1 (also called recall) is the harmonic mean of precision and hit rate.

Overall, the AUC of the logit model both in the training as well as test data set is relatively high, with a value of 84.40% and 80.10% respectively. However, looking at the confidence intervals of the AUC, it must be noted that the value may vary substantially, especially for the test data set. Here we see a 95% confidence interval of [73.50, 86.70], which seems to be a wide spread. The question remains, as to whether other methods can improve this obtained accuracy and AUC and provide tighter confidence intervals. In this respect, DTR and XGBoost may offer improvements.

5.5 DTR Results

As mentioned in the methodology, a Decision Tree Regression is investigated to determine if any improvements can be made. The resulting ROC curve of the DTR can be found in Figure 6 below. Once again, the black line represents the ROC of the DTR on the training data set and the grey line the ROC of the DTR on the test data set.

In Figure 6 the ROC curves are less smooth than those of the logit model. This possibly indicates that the predictions made by the DTR are far less varied, being rather closer to one another, than those

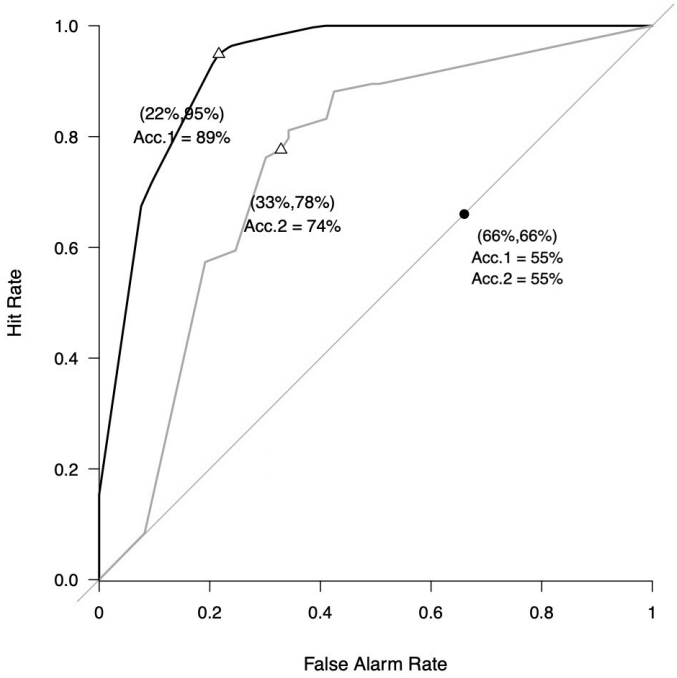

Figure 6: ROC curve for the DTR model

of the logit model. Interestingly, while the out-of-sample accuracy of the DTR model is close to that of the logit model, 74% compared to the 75% of the logit, the in-sample accuracy is substantially higher for the DTR. With an accuracy of 89% for the training data set, this is higher than the 76% achieved by the logit model. This is an indication of possible overfitting within the model. In order to avoid overfitting, as addressed in the Methodology section, two main parameters can be adjusted, the maximum depth and the minimum split. However, varying the maximum depth between 5 and 10 saw no substantial improvement in accuracy. When the maximum depth was set to an extremely low number, such as 2, the out of sample accuracy did improve slightly to 76% but this parameter choice was not further pursued as a maximum depth of 2 provides little insight on the relationship between variables as it is too small a tree. Ultimately, a maximum depth of 6 was chosen for further analysis. Adjusting the minimum split variable had even less of an effect in minimizing overfitting and improving the forecasts. The variable was assigned values between 5 and 20 in increments of 5 and no improvement was found over the original chosen value of 10. What the ROC curve indicates is that the DTR may be too rigid in its regression and it is worthwhile to investigate other methods. This is where XGBoost plays a potentially important role. Due to XGBoost using boosting through analysing various trees, it may offer more promising results than a single DTR. More on this is discussed in the latter part of the results.

Turning to the performance of the DTR model, Table 10 takes a closer look at the model's overall performance.

We can again see the difference in performance between in-sample and out of sample. Both the 50% cut-off and 66% cut-off have an in-sample accuracy of roughly 89%. This is substantially different to the out-of-sample accuracy of roughly 75% and 74% of the 50% and 66% cut-offs respectively. Furthermore,

what these accuracies show the reader is this pronounced 'grouping' or lack of variation in the DTR values. Where in the logit more variation in the accuracies is seen depending on the choice of cut-off, there seems to be little difference within the accuracy of the DTR, especially with the in-sample predictions. Looking at the 95% confidence intervals for accuracy, as with the logit, a wide spread is observed. However, what the interval does show is that the DTR is an improvement of logit for in-sample predictions. The DTR confidence intervals for both cut-offs in the training data set are above, and do not overlap, with those in the logit model. This indicates an improvement for in-sample predictions using the DTR. As with the logit, the confidence intervals for the test data set are wide for both cut-offs. Both cut-offs in the DTR, however, have similar confidence intervals whereas in the logit the cut-off of 66% had a much wider confidence interval than the 50% cut-off. This further highlights the 'grouping' of predictions previously discussed.

Table 10: Performance of DTR model on training (1) and test (2) dataset

Dataset	Cut-off	Accuracy	Acc. CI	Hit rate	FAR	Precision	F1	AUC	AUC CI
1	50.00	89.62	[84.10, 91.01]	96.37	23.53	88.86	92.46	92.60	[90.00, 95.10]
	66.00	89.26	[82.52, 91.45]	94.88	21.64	89.49	92.11		
2	50.00	75.70	[61.03, 82.02]	80.85	34.25	82.01	81.43	74.90	[67.40, 82.30]
	66.00	74.07	[60.49, 81.67]	77.62	32.88	82.22	79.86		

Note. All values are in percentages (%). F1 (also called recall) is the harmonic mean of precision and hit rate.

As to be expected, with an increased in-sample accuracy, the AUC of the DTR is also above that of the logit, with an AUC of 92.60% compared to the logit AUC of 84.40%. Furthermore, the confidence interval of the DTR is above that of the logit, showing a likely improvement. Despite this improvement of AUC, out of sample AUC saw a drop when comparing DTR to logit. The DTR model provided an out-of-sample AUC of 74.90%, substantially lower than the logit's out-of-sample AUC of 80.10%. Additionally, the confidence interval decreased and also widened, indicating a weaker performance. Where DTR did see an improvement was all values for precision besides out-of-sample using a cut-off of 66%. This improvement in precision comes from both the sharp improvement for in sample hit rate and a lower FAR, especially with the in-sample 50% cut-off.

Lastly, to investigate the relationship between the binary dependant variable and the other explanatory variables, a portion of the Decision Tree Regression plot is highlighted in Figure 7 below.

Figure 7 illustrates a portion of the 4th, 5th and 6th levels of the decision tree. Here we see the relationship of a few of the explanatory variables in classifying the dependant variable. As an example, we see the node with the decision rule 'comma' <= 7.36 on the left in the 4th level. In this case, all 26 observations, denoted as 'sample', are split according to their value of 'comma'. All values for which the statement 'comma' <= 7.36 holds enter the left node and the rest enter the right node. Looking at these two nodes we see that the left node has a 'value' equal to 0.438, indicating that the observations are closer to left-wing (which has a value of 0). Conversely, the 'value' in the right node is equal to 0.9,

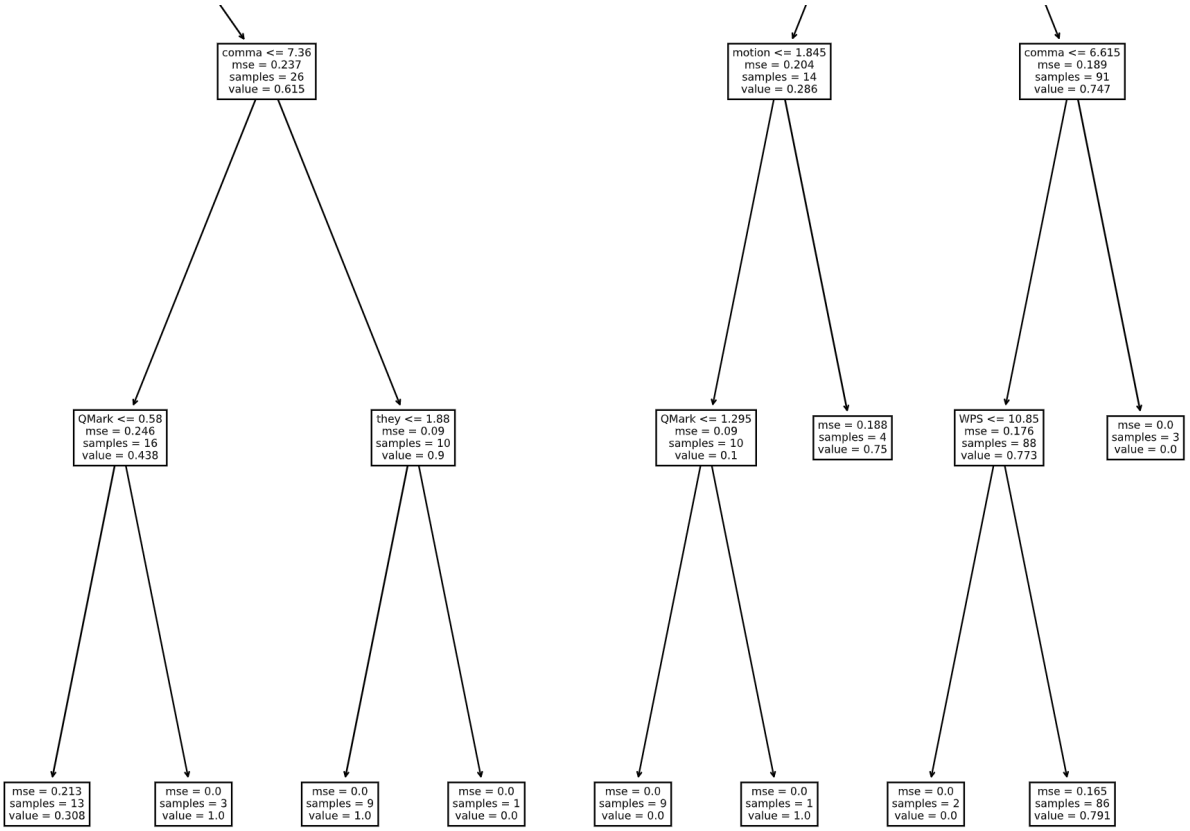

Figure 7: Excerpt of DTR plot using a maximum depth of 6 and minimum split of 10

meaning the 10 observations have a value much closer to the right-wing value of 1. In this manner, we see how the DTR builds its predictions and how they are assigned an appropriate value.

The full DTR plot, showing all six levels can be found in the Appendix in Figure 11. In total, there are 30 leaves within this decision tree. The final number of observations in each leaf varies substantially with one leaf having 186 observations in it, a fairly high number. This could be indicative that other explanatory variables may aid in expanding such heavily weighted leaves.

Despite DTR improving the in-sample predictions, there seems to be no improvement made in out-of-sample accuracy. Due to the lack of smoothness in the ROC curve for DTR, it could be beneficial to investigate using multiple trees to try to refine the predictions. This is where XGBoost could be of use, specifically in out-of-sample accuracy.

5.6 XGBoost Results

Figure 8 shows the ROC curve for XGBoost on the training and test data. As expected, the ROC curves for both in-sample as well as out of sample data sets result in much smoother lines than that of DTR. This is indicative that XGBoost can provide more varied predictions than DTR and when varying cut-offs are used, there is no sharp change. This already shows an improvement over the DTR.

Looking at the ROC of the training data, we see an improvement over the logit ROC in terms of accuracy. Compared to the logit's in sample accuracy of 76%, XGBoost has an in-sample accuracy of 87%. Once again, this may be an indication of over-fitting although the accuracy is lower than the 89%

accuracy of the DTR. To address over-fitting we look at the ROC curve using the test data set, the grey line. With an out-of-sample accuracy of 76%, XGBoost has the highest out-of-sample accuracy of all three models. Albeit a singular percentage point higher than the logit model, it is also an improvement over the poorer performance of the DTR. With the out-of-sample accuracy and in-sample accuracy bridging the gap, there seems to be a reduction in over-fitting going from DTR to XGBoost. Three additional parameters were used in the modelling of XGBoost to reduce this over-fitting. These were; the percentage of features used per tree (colsample_bytree = 0.3), the step-size shrinkage (learning_rate = 0.1) and the regularization of leaf weights (alpha = 9).

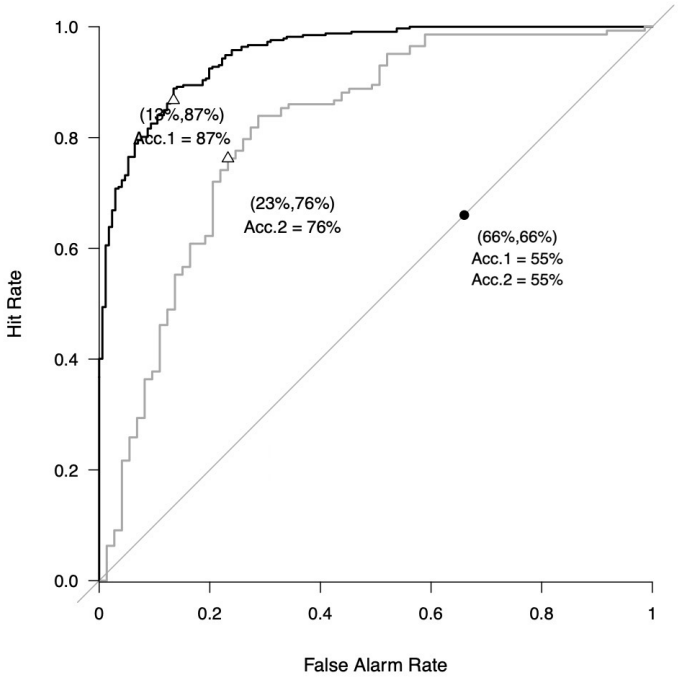

Figure 8: ROC curve for the XGB model

While the ROC curve for XGBoost indicates an improvement over the logit and DTR ROC curves, a further look is needed. Table 11 shows additional insight into the performance of XGBoost. Looking at the training data set, we see an improvement in accuracy compared to logit for both cut-offs. This is substantiated by the 95% confidence intervals, both of which are higher than those of the logit. The confidence interval of the 50% cut-off has also narrowed compared to the same confidence interval in the logit. Furthermore, the hit rate for both has improved. The FAR has decreased compared to logit but, as can be expected, neither perform better than the DTR. Where XGBoost does outperform both the logit and DTR is in AUC with a value of 95.10%. However, it must be noted that although the confidence interval for this AUC has narrowed compared to logit and DTR, it still overlaps with the respective confidence interval in DTR, implying that it may not be a clear improvement. Interestingly, in terms of precision, XGBoost and DTR alternate which is better depending on the cut-off, indicating similar performance. Both outperform the logit model in terms of precision and F1.

Turning to out of sample performance, it is noted that the accuracy is the same for both cut-off

Table 11: Performance of XGB model on training (1) and test (2) dataset

Dataset	Cut-off	Accuracy	Acc. CI	Hit rate	FAR	Precision	F1	AUC	AUC CI
1	50.00	87.28	[85.69, 88.47]	97.59	32.75	85.26	91.01	95.10	[93.30, 96.80]
	66.00	86.68	[82.31, 90.46]	86.75	13.45	92.60	89.58		
2	50.00	76.39	[72.22, 82.41]	90.21	50.68	77.71	83.50	81.30	[74.80, 87.80]
	66.00	76.39	[61.11, 84.26]	76.22	23.29	86.51	81.04		

Note. All values are in percentages (%). F1 (also called recall) is the harmonic mean of precision and hit rate.

choices and outperforms the logit and DTR. Looking at the confidence intervals, we see that with a 50% cut-off a similar interval is found as with logit, improving upon the interval of the DTR. Nevertheless, little improvement is shown in the interval for the 66% cut-off. Not only is accuracy higher in XGBoost but with an AUC of 81.30%, it offers the highest AUC over all three models. The AUC has a confidence interval similar to logit which is to be expected due to the close AUCs. This underscores that although the achieved AUC is higher, it remains to be discussed whether this improvement is meaningful, due to the closeness of the respective confidence intervals. Additionally, where the model seems to struggle slightly is in the FAR. With a cut-off of 50%, XGBoost has an out-of-sample FAR of 50.68%, the highest value so far. However, this large FAR value quickly drops with a cut-off of 66%. This observation highlights the importance of the chosen cut-off. While both provide the same accuracy, there is a large trade-off between Hit rate and FAR for the two choices. The F1 for out-of-sample predictions is an improvement to the values obtained with DTR. Interestingly the performance of the F1 statistic between logit and XGBoost depends on the choice of cut-off, likely due to the large FAR value in XGBoost with a cut-off of 50%.

One useful aspect of XGBoost is its ability to report the most important features of its model. It does so by looking at each variable and determining how much it has contributed to improving the performance of the model through an F-score. More specifically, within each tree, it looks at the various splits of the data at each node. As seen in the DTR which displays a single tree, at each node a rule is constructed based on one specific explanatory variable. XGBoost looks at all these splits and determines how much the performance of its model increases with each of these rules through the use of an F-score. The improvement is then weighted according to the number of observations within the node. By analysing each node in the tree, the relative importance of each utilized explanatory variable is given. XGBoost then averages these importance scores over all the trees. In this analysis, 1000 trees were used to create the model. Although it was found that 1000 trees provided no improvement of the model compared to using 100 in terms of accuracy, 1000 trees were still chosen. This is due to the fact that with a larger amount of trees, the relative importance of each feature, shown in Figure 9, is built over many more iterations. This may add a level of robustness as to which features are actually the most persistently useful for the model.

Looking closer at the relative importance of all the features within the XGBoost model, an interesting

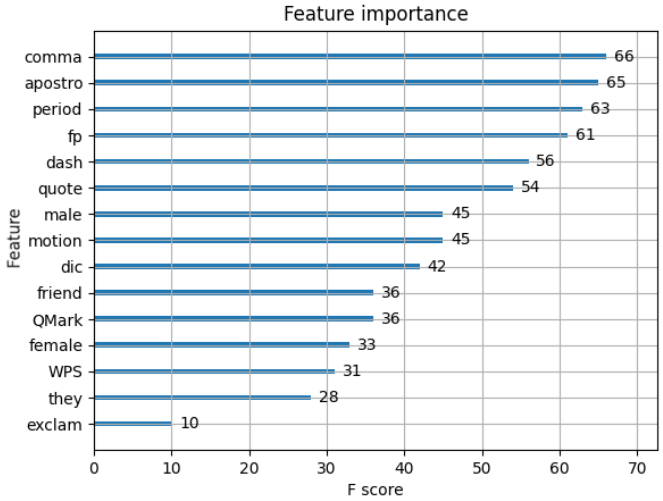

Figure 9: Importance of each feature when training XGBoost. fp represents the variable 'focuspresent'

observation is found in the first three variables. The three most important features all relate to forms of punctuation. Commas, apostrophes and periods are the most persistently important features in the XGBoost model, indicating that punctuation is more indicative of political orientation than language use, a surprising result. Moving down the graph, language use starts to appear through variables such as 'focus present' (fp). However, we still see important punctuation variables such as dashes and quotation marks ranking among the most important variables for XGBoost. Lastly, more emotive variables such as exclamation marks and 'grouping' words such as 'they' or 'friend' rank surprisingly low on the importance chart. This counters the point made earlier that right wing news tends to use more emotive and grouping language in their texts. While still important, their ranking shows a clear drop-off in usefulness within the model compared to punctuation variables.

One comparison of interest is the difference between the importance feature graph and the marginal effects of the variables in the logit model, as seen in Table 8. In the logit model, the variables *Exclam* and *friend* had the two largest absolute marginal effects yet interestingly in the Feature Importance graph, both show a relatively low F-score. This highlights the different approaches of both models. Where in the logit *Exclam* has one of the largest marginal effects on the dependent variables, in XGboost *Exclam* seems to have a fairly low F-score and hence a low accuracy improvement for the model. Similarly the variable *comma*, which is ranked the most important in Figure 9, has the fifth-lowest absolute marginal effect in the logit model. These results highlight the benefits of investigating different models and seeing the varying usefulness of each explanatory variable within its respective model.

Lastly, as was done with the DTR, we examine one of the trees in the fully boosted XGBoost model. Since XGBoost makes use of boosting and this paper uses 1000 trees in its model building, the focus will be given to the first tree in the fully boosted model. This tree can be found in Figure 10 below.

Figure 10 has several differences to the DTR tree. Due to XGBoost using over 1000 trees, the XGBoost tree is not exhaustive as with DTR. This can be seen with the maximum depth. The maximum depth for XGBoost is set to 7 while this tree only has a maximum depth of 4.

Figure 10: First Tree in XGB Model

Figure 10 highlights the prominence of the 'period' variable in classifying data. Interestingly, the tree also utilizes the three least important features; 'exclam', 'WPS' and 'they'. As it is the first tree it may be that these three variables play a role here but their importance diminishes as more trees are created. Here, as with the DTR plot, we see the nodes split in the same manner. Observations are split according to a threshold for an explanatory variable and then assigned a value corresponding to its predicted political orientation.

5.7 Common Correct Predictions

Now that the three models have been analysed and their performance assessed, it remains to examine their commonality in terms of their correct predictions. While logit is a generalized linear regression, DTR and XGBoost are non-linear and make use of a tree structure. It could be that using a non-linear model results in vastly different correct predictions than that of a linear one even though the accuracies may be close. A common correct prediction is simply when both models correctly classify an observation as left or right wing. As seen when comparing the Feature Importance graph from XGBoost and the marginal effects table for the logit model, models may utilize explanatory variables and weigh their effects differently. This could lead to divergent predictions and, in order to compare common performance, the common correct predictions are further investigated in this Section. To compare the predictions, Tables 12 to 15 are constructed. Table 12 and 13 show the proportion of similar correct predictions for the training data, using 50% and 66% cut-off respectively.

Table 12: Proportion of same correct predictions for training data using 50% cut-off

	Logit	DTR	XGB
Logit	1	0.802	0.859
DTR	0.802	1	0.913
XGB	0.859	0.913	1

Table 13: Proportion of same correct predictions for training data using 66% cut-off

	Logit	DTR	XGB
Logit	1	0.746	0.831
DTR	0.746	1	0.841
XGB	0.831	0.841	1

When looking at the in-sample accuracy of all three models, it was found that DTR had the highest accuracy followed by XGBoost and then logit. This seems to be reflected here in the shared proportion of correct predictions. For both the 50% and 66% cut-offs, DTR and logit share the lowest proportion of common correct predictions, with 80.2% and 74.6% respectively. This is to be expected since DTR has the highest in-sample accuracy and logit the lowest. Under the same logic, it is not surprising that XGBoost and DTR share a higher proportion of correct predictions than either have with logit. However, what is of interest is the large decrease in the difference between proportions when the cut-off is changed. DTR and XGboost have a common proportion of 91.3% with a cut-off of 50% and XGBoost and logit a common proportion of 85.9%. When the cut-off is changed to 66%, this difference becomes much smaller, only differing by one percentage point. What these results show is that the proportion of common correct predictions is fairly inline with the observed accuracies seen earlier in this paper. It additionally highlights, however, the sensitivity of these proportions according to the chosen cut-offs. Moving onto the proportion of similar correct predictions for the test data set, Tables 14 and 15 give additional insight into the commonality of predictions.

Table 14: Proportion of same correct predictions for test data using 50% cut-off

	Logit	DTR	XGB
Logit	1	0.779	0.820
DTR	0.779	1	0.788
XGB	0.820	0.788	1

Table 15: Proportion of same correct predictions for test data using 66% cut-off

	Logit	DTR	XGB
Logit	1	0.760	0.820
DTR	0.760	1	0.779
XGB	0.820	0.779	1

Compared to those of the training set, we mostly see a reduction in common correct predictions – with one exception being logit and DTR with a 66% cut-off. A second observation is that the highest proportion of common correct predictions is no longer XGB and DTR but rather XGB and logit. This is due to XGB and logit having a higher out-of-sample accuracy than DTR. The drop in common correct prediction for DTR and XGBoost may be evidence of a reduction in over-fitting in XGB and a better out-of-sample performance than the DTR. XGB performs well in the training data, much like DTR, and therefore has the highest proportion of common predictions with DTR. However, when looking at the out-of-sample predictions, where logit performs better than DTR, we notice XGB and logit sharing the highest proportion of common correct predictions.

This may indicate that XGBoost is a strong middle ground between logit and DTR, using tree structures to improve in-sample predictions and boosting to improve out of sample predictions.

What the previous four tables have shown is that the predictions of the three models follow the expected order given by their accuracy rankings. Models with higher accuracy tend to share a higher proportion of correct predictions. This implies that the models, although constructed differently, do not offer vastly different predictions. This may further imply that the explanatory power of the LIWC variables only has a limited use in prediction. Perhaps a further improvement in accuracy may not come from new or improved models but rather by including additional explanatory variables.

6 Limitations and further research

One limitation within this paper is the constructed data set. As with most data sets concerning veracity or political orientation, it may contain inherent biases. Due to the fact that these data sets have been labelled and compiled by humans, the classification for each article, whether it is veracity or the political orientation, is dependant on the reviewer's values. The journalists who created this data set have employed several checks and balances such as using multiple reviewers but this limitation is still important to mention. A second point is the choice of hyper-parameters for the DTR and XGBoost. Literature was followed to set the initial hyper-parameters but these models have a wide arrange of optionality concerning hyper-parameter choices. To strengthen the robustness of the results obtained in this paper, a deeper dive into hyper-parameter tuning may be worth investigating. Additionally, additional hyper-parameter tuning may even provide improved accuracy than what is achieved in this paper.

Another facet of further research that may be of interest is exploring additional LIWC variables from older dictionaries. In this paper, the 2015 LIWC dictionary is used to construct all LIWC variables. However, a 2001 and 2007 dictionary also exists and could be insightful for further research. As mentioned in the section concerning common correct predictions, additional explanatory variables could provide higher accuracies and improve the models. This is primarily due to the fact that the accuracy of all three investigated models is close, meaning the LIWC variables only have so much explanatory power. Since the focus of this paper is the use of language variables in political orientation classification, other explanatory variables are not pursued. However, less conventional language variables such as the observed sentiment variable may be useful for further research. The research in this paper has made exciting strides in developing a model for classifying the political orientation of articles. With the base of these models built upon a paper seeking to classify the veracity of articles van der Zee et al. (in press), this thesis is a testament to the wide scope language models can have in classifying content. This paper urges for further research into the usefulness of language analysis in content classification and hopes the promising results here spark interest in the other uses of language analysis in fields besides deception and political orientation detection.

7 Conclusion and Discussion

H1 put forward the idea that true and false articles will have statistically significant linguistic differences. This was due to the wide array of research surrounding language and deception detection which had found similar results. To test this hypothesis a MANOVA, as in van der Zee et al. (in press), was run. Interestingly, however, and in contrast to previous literature, little evidence was found of statistically significant linguistic differences within this paper's data set. Perhaps this is due to the nature of the articles. All articles analysed here are hyper-partisan and may already be very emotive, regardless of the degree of factualness in the text. This is supported by the previously analysed sentiment data, which indicates that both left wing and right wing articles tend to be much more negative than mainstream news. It may be that due to this emotive nature of the articles, that the linguistic cues do not change as much as is expected when lies are told. The theory previously discussed, where liars find themselves

in different cognitive states compared to truth-tellers may have less validity in hyper-partisan news. Due to the already emotive state authors are in when discussing their political outlooks, the veracity of their statements may have less of a cognitive effect. The results obtained through this MANOVA are surprising and a possible point of extension for those exploring deception detection, as it starkly contrasts with the findings of previous deception detection research using linguistic analysis. It may be that deception detection using linguistic analysis is not as insightful when used for emotive texts. However, it must be noted that an FDR correction was used for the p-values in the MANOVA, making for conservative results. Nevertheless, as mentioned in the Results section of this paper, the same approach was used by van der Zee et al. (in press) who indeed did find statistically significant linguistic differences depending on the veracity of articles.

Where statistically significant linguistic differences did appear was in the analysis of left and right wing news. In order to test H2, the same MANOVA approach was taken. Of the 82 analysed variables, the MANOVA returned 25 statistically significant linguistic variables even when using an FDR correction. Interestingly, right-wing news tended to have a higher proportion of what seems like 'grouping' words as well as a higher proportion of emotive words. These results offer promising insights for the use of language variables in political orientation classification and support the hypothesis put forward that left and right wing authors have different language usage.

Using the statistically significant linguistic variables, three models are put forward for predicting political orientation; logistic, DTR and XGBoost. To test H3, which states that political ideology classification methods using linguistic variables will outperform random guessing models, the ROC and performance tables are examined. The DTR provided the highest in-sample accuracy of 89.62%. The logit and XGBoost shared the position of highest out of sample accuracy depending on the cut-off value. With a 50% cut-off logit had the highest out-of-sample accuracy of 78.24%. Conversely, with a cut-off of 66% XGBoost had the highest out-of-sample accuracy of 76.39%. Either way, both models shared a similar overall performance in the out of sample data, with a large overlap in confidence intervals for both accuracy and AUC. All three models do provide an improvement over random guessing. All three ROC curves, either in the test or training set are above the diagonal ROC curve for random guessing. While an improvement is seen over random guessing, this paper suggests further research in minimizing the over-fitting found in the two tree models and perhaps using additional explanatory variables to increase accuracy even further.

This paper seeks to investigate the relationship of language-based variables with political orientation and veracity. While the results show a minimal relationship between the latter, the former provides a promising outlook on using linguistic variables for political ideology identification. Three models have outperformed random guessing and lay the foundation for further investigation into the usefulness of language analysis for the automatic political classification of texts.

8 Bibliography

References

Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, *31*(2), 211-236.

Balz, D. (2016, Jan). *A divided country gets a divisive election.* Washington Post. Retrieved 07-05-2021, from https://www.washingtonpost.com/politics/a-divided-country-gets-a-divisive-election/2016/01/09/591bfccc-b61f-11e5-a842-0feb51d1d124_story.html

Brownlee, J. (2018, August). *How to use roc curves and precision-recall curves for classification in python.* Machine Learning Mastery. Retrieved 19-05-2021, from https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/

Conroy, N. K., Rubin, V. L., & Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, *52*(1), 1-4.

Crowell, C. (2017, June). *Our approach to bots and misinformation.* Twitter. Retrieved 07-05-2021, from https://blog.twitter.com/en_us/topics/company/2017/Our-Approach-Bots-Misinformation.html

Cutler, A. D., Carden, S. W., Dorough, H. L., & Holtzman, N. S. (2020). Inferring Grandiose Narcissism from text: LIWC versus machine learning. *Journal of Language and Social Psychology*, *40*(2), 260–276.

Facebook. (2017, April). *Working to stop misinformation and false news.* Author. Retrieved 07-05-2021, from https://www.facebook.com/formedia/blog/working-to-stop-misinformation-and-false-news

Fuller, C., Biros, D., Twitchell, D., & Wilson, R. (2015). Real-World Deception and the Impact of Severity. *Journal of Computer Information Systems*, *55*(2), 59–67.

Giolla, E. M., Granhag, P. A., & Vernham, Z. (2017). Drawing-based deception detection techniques: A state-of-the-art review. *Crime Psychology Review*, *3*(1), 23–38.

Hauch, V., Blandón-Gitlin, I., Masip, J., & Sporer, S. L. (2015). Are computers effective lie detectors? A meta-analysis of linguistic cues to deception. *Personality and Social Psychology Review*, *19*(4), 307–342.

Hoffstein, C. (2020, May). *Flirting with models.* The Research Library of Newfound Research. Retrieved 19-05-2021, from https://blog.thinknewfound.com/2020/05/defensive-equity-with-machine-learning/xg-boost-final-01/

Hutto, C., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text in. *Proceedings of the 8th International Conference on Weblogs and Social Media, ICWSM 2014*, Ann Arbor, Michigan, United States.

Instagram. (2019, Dec). *Combatting Misinformation on Instagram.* Author. Retrieved 07-05-2021, from https://www.pbs.org/newshour/science/real-consequences-fake-news-stories-brain-cant-ignore

Jurkowitz, M., Mitchell, A., Shearer, E., & Walker, M. (2021, Jan). *U.s. media polarization and the 2020 election: A nation divided.* Pew Research Center's Journalism Project. Retrieved 08-05-2021, from https://www.journalism.org/2020/01/24/u-s-media-polarization-and-the-2020-election-a-nation-divided/

Koppel, M., Schler, J., & Bonchek-Dokow, E. (2007). Measuring Differentiability: Unmasking Pseudonymous Authors. *Journal of Machine Learning Research*, *8*(6), 1261–1276.

Lewis, R. J. (2000). An introduction to classification and regression tree (CART) analysis. *Annual meeting of the Society for Academic Emergency Medicine*, San Francisco, California, United States.

Mantovani, R. G., Horváth, T., Cerri, R., Junior, S. B., Vanschoren, J., & de Carvalho, A. C. P. d. L. F. (2018). An empirical study on hyperparameter tuning of decision trees. *Corr*, *arXiv: 1812.02207v2*.

Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2017). A stylometric inquiry into hyperpartisan and fake news. *arXiv:1702.05638*.

Reis, J. C., Correia, A., Murai, F., Veloso, A., & Benevenuto, F. (2019). Supervised learning for fake news detection. *IEEE Intelligent Systems*, *34*(2), 76–81.

Rouam, S. (2013). *Encyclopedia of systems biology.* New York, NY: Springer.

Sherr, I. (2018, April). *Facebook, Cambridge Analytica, data mining and TRUMP: What you need to know.* CNET. Retrieved 07-05-2021, from https://www.cnet.com/news/facebook-cambridge-analytica-data-mining-and-trump-what-you-need-to-know/

Silverman, C., Strapagiel, L., Shaban, H., Hall, E., & Singer-Vine, J. (2016, Oct). *Hyperpartisan Facebook pages are publishing false and misleading information at an alarming rate.* BuzzFeed. Retrieved 08-05-2021, from https://www.buzzfeednews.com/article/craigsilverman/partisan-fb-pages-analysis

van der Zee, S., Poppe, R., Havrileck, A., & Baillon, A. (in press). A personal model of trumpery: Linguistic deception detection in a real-world high-stakes setting. *Psychological Science*.

9 Appendix

Table 16: Table 1/3 with MANOVA results comparing Right and Left wing texts

Variable	Mean_if_Left	Mean_if_Right	F_stat.	p-value	Sig.	Bayes Factor	Cohen's d	CI
adj	4.21	4.13	0.34	0.760		0.120	0.05	[-0.13,0.24]
adverb	3.99	3.96	0.05	0.910		0.110	0.02	[-0.16,0.20]
affect	5.11	5.32	1.34	0.442		0.200	-0.11	[-0.29,0.08]
affiliation	1.72	2.01	5.58	0.056		1.540	-0.22	[-0.41,-0.04]
AllPunc	16.00	16.89	4.03	0.109		0.730	-0.19	[-0.37,0.00]
Analytic	79.74	76.99	3.51	0.137		0.570	0.18	[-0.01,0.36]
anger	1.21	1.10	1.38	0.438		0.200	0.11	[-0.07,0.30]
anx	0.41	0.45	0.55	0.676		0.140	-0.07	[-0.25,0.12]
Apostro	1.51	2.18	28.75	0.000	***	>1,000	-0.50	[-0.69,-0.32]
article	7.39	7.66	2.55	0.233		0.360	-0.15	[-0.34,0.03]
assent	0.12	0.11	0.24	0.760		0.120	0.05	[-0.14,0.23]
Authentic	15.58	18.97	5.84	0.051		1.750	-0.23	[-0.41,-0.04]
auxverb	7.75	8.01	1.86	0.346		0.260	-0.13	[-0.31,0.06]
cause	1.59	1.55	0.20	0.771		0.110	0.04	[-0.14,0.23]
certain	1.61	1.43	4.96	0.072		1.140	0.21	[0.02,0.39]
Clout	72.31	74.69	5.26	0.063		1.320	-0.22	[-0.40,-0.03]
cogproc	9.98	10.10	0.20	0.771		0.110	-0.04	[-0.23,0.14]
Colon	0.46	0.35	6.50	0.038	*	2.390	0.24	[0.05,0.43]
Comma	4.91	4.34	14.82	0.001	**	126.500	0.36	[0.18,0.55]
compare	2.31	2.18	1.67	0.374		0.230	0.12	[-0.06,0.31]
compound	-0.29	-0.22	0.87	0.576		0.000	-0.09	[-0.27,0.10]
conj	5.19	5.55	7.54	0.026	*	3.940	-0.26	[-0.44,-0.07]
Dash	0.85	0.58	18.95	0.000	***	885.120	0.41	[0.22,0.60]
Dic	78.18	80.92	26.19	0.000	***	>1,000	-0.48	[-0.67,-0.29]
differ	2.71	2.82	1.01	0.539		0.170	-0.09	[-0.28,0.09]
discrep	1.23	1.32	1.17	0.489		0.180	-0.10	[-0.29,0.08]
drives	8.29	8.73	3.63	0.131		0.600	-0.18	[-0.36,0.01]
Exclam	0.09	0.25	10.50	0.008	**	16.200	-0.30	[-0.49,-0.12]
family	0.22	0.34	4.92	0.072		1.120	-0.21	[-0.39,-0.02]
feel	0.35	0.36	0.01	0.923		0.110	-0.01	[-0.20,0.17]
female	0.49	1.28	26.74	0.000	***	>1,000	-0.49	[-0.67,-0.30]
filler	0.01	0.01	0.02	0.912		0.110	-0.01	[-0.20,0.17]
focusfuture	0.92	0.96	0.25	0.760		0.120	-0.05	[-0.23,0.14]

Note. * p <0.05, ** p <0.01, *** p <0.001

Table 17: Table 2/3 with MANOVA results comparing Right and Left wing texts

Variable	Mean_if_Left	Mean_if_Right	F_stat.	p-value	Sig.	Bayes Factor	Cohen's d	CI
focuspast	4.37	4.48	0.33	0.760		0.120	-0.05	[-0.24,0.13]
focuspresent	7.81	8.51	6.99	0.031	*	3.030	-0.25	[-0.43,-0.06]
friend	0.12	0.21	10.18	0.008	**	13.940	-0.30	[-0.49,-0.11]
function	46.92	48.67	11.75	0.005	**	29.460	-0.32	[-0.51,-0.14]
hear	0.91	0.92	0.01	0.923		0.110	-0.01	[-0.20,0.17]
i	0.61	0.87	7.25	0.029	*	3.440	-0.25	[-0.44,-0.07]
informal	0.71	0.64	0.54	0.676		0.140	0.07	[-0.12,0.25]
insight	1.97	2.04	0.42	0.724		0.130	-0.06	[-0.25,0.12]
interrog	1.57	1.57	0.00	0.955		0.100	0.01	[-0.18,0.19]
ipron	5.16	5.37	1.60	0.384		0.230	-0.12	[-0.30,0.07]
male	2.54	1.77	20.94	0.000	***	>1,000	0.43	[0.24,0.62]
motion	1.36	1.62	11.99	0.005	**	32.980	-0.33	[-0.51,-0.14]
negate	1.51	1.49	0.07	0.894		0.110	0.03	[-0.16,0.21]
negemo	2.67	2.73	0.18	0.775		0.110	-0.04	[-0.22,0.14]
netspeak	0.32	0.30	0.06	0.896		0.110	0.02	[-0.16,0.21]
nonflu	0.12	0.12	0.03	0.912		0.110	-0.02	[-0.20,0.17]
number	2.13	1.79	5.30	0.063		1.340	0.22	[0.03,0.40]
OtherP	0.84	0.64	2.43	0.245		0.340	0.15	[-0.04,0.33]
Parenth	0.66	0.48	3.67	0.131		0.610	0.18	[-0.01,0.37]
percept	2.54	2.64	0.52	0.678		0.130	-0.07	[-0.25,0.12]
Period	4.91	5.62	27.82	0.000	***	>1,000	-0.50	[-0.68,-0.31]
posemo	2.40	2.52	0.85	0.576		0.160	-0.09	[-0.27,0.10]
power	4.20	4.24	0.07	0.894		0.110	-0.02	[-0.21,0.16]
ppron	5.26	6.05	10.03	0.008	**	12.980	-0.30	[-0.48,-0.11]
prep	13.78	13.81	0.03	0.912		0.110	-0.02	[-0.20,0.17]
pronoun	10.43	11.42	9.17	0.012	*	8.600	-0.28	[-0.47,-0.10]
QMark	0.26	0.39	6.93	0.031	*	2.950	-0.25	[-0.43,-0.06]
quant	1.85	1.90	0.29	0.760		0.120	-0.05	[-0.24,0.13]
Quote	1.48	2.05	11.50	0.005	**	26.190	-0.32	[-0.51,-0.13]
relativ	12.64	12.82	0.41	0.724		0.130	-0.06	[-0.25,0.12]
reward	1.08	1.14	0.31	0.760		0.120	-0.05	[-0.24,0.13]
risk	0.76	0.81	0.67	0.630		0.140	-0.08	[-0.26,0.11]

Note. * p <0.05, ** p <0.01, *** p <0.001

Table 18: Table 3/3 with MANOVA results comparing Right and Left wing texts

Variable	Mean_if_Left	Mean_if_Right	F_stat.	p-value	Sig.	Bayes Factor	Cohen's d	CI
sad	0.30	0.33	0.67	0.630	0.140	-0.08	[-0.26,0.11]	[-0.24,0.13]
see	1.21	1.30	0.86	0.576	0.160	-0.09	[-0.27,0.10]	[-0.43,-0.06]
SemiC	0.04	0.03	0.26	0.760	0.120	0.05	[-0.14,0.23]	[-0.49,-0.11]
shehe	2.59	2.49	0.28	0.760	0.120	0.05	[-0.14,0.23]	[-0.51,-0.14]
Sixltr	23.46	22.38	6.29	0.041	*	2.160	0.24	[0.05,0.42]
social	9.99	10.91	8.55	0.016	*	6.400	-0.28	[-0.46,-0.09]
space	6.54	6.78	1.82	0.348	0.250	-0.13	[-0.31,0.06]	[-0.12,0.25]
swear	0.13	0.10	2.66	0.223	0.380	0.15	[-0.03,0.34]	[-0.25,0.12]
tentat	2.01	2.25	4.86	0.072	1.090	-0.21	[-0.39,-0.02]	[-0.18,0.19]
they	0.83	1.16	13.79	0.002	**	77.530	-0.35	[-0.54,-0.16]
time	4.82	4.45	4.69	0.076	1.010	0.20	[0.02,0.39]	[0.24,0.62]
Tone	29.95	30.34	0.02	0.912	0.110	-0.01	[-0.20,0.17]	[-0.51,-0.14]
verb	13.71	14.59	10.12	0.008	**	13.510	-0.30	[-0.49,-0.11]
WC	416.16	424.56	0.21	0.771	0.120	-0.04	[-0.23,0.14]	[-0.22,0.14]
we	0.68	0.74	0.79	0.589	0.150	-0.08	[-0.27,0.10]	[-0.16,0.21]
WPS	22.69	19.39	35.86	0.000	***	>1,000	0.56	[0.38,0.75]
you	0.56	0.78	8.84	0.014	*	7.360	-0.28	[-0.47,-0.09]

Note. * p <0.05, ** p <0.01, *** p <0.001

Table 19: LIWC or other key term and corresponding variable name

Variable Name	LIWC/Key name	Variable Name	LIWC/Key Name
Adjectives	adj	Impersonal pronouns	ipron
Adverbs	adverb	Male references	male
Emotions	affect	Motion	motion
Affiliation	affiliation	Negations	negate
All punctuation	AllPunc	Negative emotions	negemo
Analytic thinking	Analytic	Netspeak	netspeak
Anger	anger	Nonfluencies	nonflu
Anxiety	anx	Numbers	number
Apostrophes	Apostro	Other punctuation	OtherP
Articles	article	Parentheses (pairs)	Parenth
Assent	assent	Perceptual processes	percept
Authentic	Authentic	Periods	Period
Auxiliary verbs	auxverb	Positive emotions	posemo
Causations	cause	Power	power
Certainty	certain	Personal pronouns	ppron
Clout	Clout	Prepositions	prep
Cognitive processes	cogproc	Total pronouns	pronoun
Colons	Colon	Question marks	QMark
Commas	Comma	Quantifiers	quant
Comparison words	compare	Quotation marks	Quote
Conjunctions	conj	Relativity	relativ
Dashes	Dash	Reward focus	reward
Dictionary words	Dic	Risk focus	risk
Differentiation	differ	Sadness	sad
Discrepancy	discrep	Seeing	see
Drives	drives	SemiColon	SemiC
Exclamation marks	Exclam	Third-person singular pronouns	shehe
Family	family	Six-letter words	Sixltr
Feeling	feel	Social processes	social
Female references	female	Space	space
Fillers	filler	Swear words	swear
Future orientation	focusfuture	Tentative	tentat
Past orientation	focuspast	Third-person plural pronouns	they
Present orientation	focuspresent	Time	time
Friends	friend	Emotional tone	Tone
Total function words	function.	Common verbs	verb
Hearing	hear	Word quantity	WC
First-person singular pronouns	i	First-person plural pronouns	we
Informal	informal	Average sentence length	WPS
Insight	insight	Total second-person pronouns	you
Interrogatives	interrog	Sentiment	compound

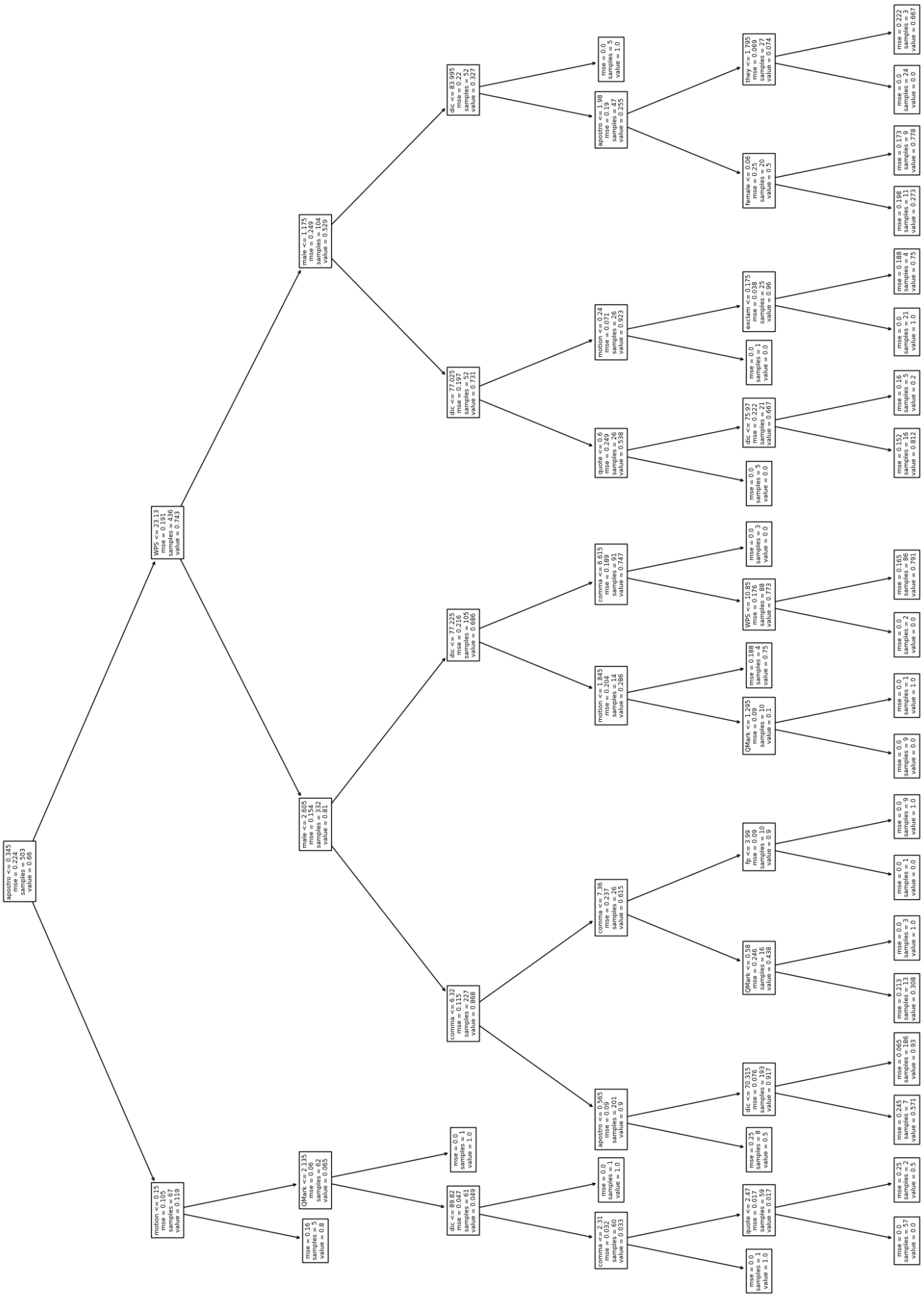

Figure 11: DTR plot using a maximum depth of 6 and minimum split of 10