Team Fernando-Pessa at SemEval-2019 Task 4: Back to Basics in Hyperpartisan News Detection

André Ferreira Cruz
Universidade do Porto
Faculdade de Engenharia
andre.ferreira.cruz@fe.up.pt

Gil Rocha
Universidade do Porto
Faculdade de Engenharia
LIACC
gil.rocha@fe.up.pt

Rui Sousa-Silva
Universidade do Porto
Faculdade de Letras
CLUP
rssilva@letras.up.pt

Henrique Lopes Cardoso
Universidade do Porto
Faculdade de Engenharia
LIACC
hlc@fe.up.pt

Abstract

This paper describes our submission[1] to the SemEval 2019 Hyperpartisan News Detection task. Our system aims for a linguistics-based document classification from a minimal set of interpretable features, while maintaining good performance. To this goal, we follow a feature-based approach and perform several experiments with different machine learning classifiers. On the main task, our model achieved an accuracy of 71.7%, which was improved after the task's end to 72.9%. We also participate in the meta-learning sub-task, for classifying documents with the binary classifications of all submitted systems as input, achieving an accuracy of 89.9%.

1 Introduction

Hyperpartisan news detection consists in identifying news that exhibit extreme bias towards a single side (Potthast et al., 2018). The shift, in news consumption behavior, from traditional outlets to social media platforms has been accompanied by a surge of fake and/or hyperpartisan news articles in recent years (Gottfried and Shearer, 2017), raising concerns in both researchers and the general public. As ideologically aligned humans prefer to believe in ideologically aligned news (Allcott and Gentzkow, 2017), these tend to be shared more often and, thus, spread at a fast and unchecked pace. Moreover, there is a large intersection of 'fake' and 'hyperpartisan' news, as 97% of fake news articles in BuzzFeed's Facebook fact-check dataset are also hyperpartisan (Silverman et al., 2016).

However, the detection/classification and consequent regulation of online content must be done with careful consideration, as any automatic system risks unintended censorship (Akdeniz, 2010). As such, we aim for a linguistically-guided model from a set of interpretable features, together with classifiers that facilitate inspection of what the model has learned, such as Random Forests (Ho, 1995), Support Vector Machines (Cortes and Vapnik, 1995) and Gradient Boosted Trees (Drucker and Cortes, 1996). Neural network models are left out for their typically less self-explanatory nature.

The SemEval 2019 Task 4 (Kiesel et al., 2019) challenged participants to build a system for hyperpartisan news detection. The provided dataset consists of 645 manually annotated articles (*by-article dataset*), as well as 750,000 articles automatically annotated publisher-wise (*by-publisher* dataset, split 80% for training and 20% for validation). Systems are ranked by accuracy on a set of unpublished test articles (from the *by-article* dataset), which has no publishers in common with the provided train dataset, preventing accuracy gains by profiling publishers. All experiments on this paper are performed on the *gold-standard* (*by-article*) corpus, as this was the official dataset.

The rest of the paper is organized as follows. Section 2 describes our pre-processing, feature selection, and the system's architecture. Section 3 analyzes our model's performance, evaluates each feature importance, and goes in-depth on some classification examples. Finally, Section 5 draws conclusions and sketches future work.

2 System Description

We propose a feature-based approach and experiment with several machine learning algorithms, namely support vector machines with linear ker-

[1] https://github.com/AndreFCruz/semeval2019-hyperpartisan-news

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 999–1003
Minneapolis, Minnesota, USA, June 6–7, 2019.

nels (SVM), random forests (RF), and gradient boosted trees (GBT). Our submission to the task was a RF classifier, as this was the best performing at the time. However, after the task's end we found a combination of hyperparameters that turned GBT into the best-performer. We detail all results in the following sub-sections.

All classifiers were implemented using *scikit-learn* (Pedregosa et al., 2011) for the *Python* programming language, and all were trained on the same dataset of *featurized* documents. In this section we describe the data pre-processing, our selection of features, as well as the classifiers' grid-searched hyperparameters.

2.1 Feature Selection

The statistical analysis of natural language has been widely used for stylometric purposes, in particular in order to define linguistic features to measure author style. These include, among others: document length, sentence and word length, use of punctuation, use of capital letters, and frequency of word n-grams; type-token ratio (Johnson, 1944); and frequency of word n-grams (see e.g. Stamatatos (2009) for a thorough survey of authorship attribution methods). Although these features have been successfully used in authorship attribution to establish the most likely writer of a target text among a range of possible writers (Sousa-Silva et al., 2010, 2011), research on how these features can be used to analyze group authorship – and subsequently identify an ideological slant – is less demonstrated. Therefore, we build upon previous research on Computer-Mediated Discourse Analysis (Herring, 2004) to test the use of these features to detect hyperpartisan news.

We compute a minimal set of style and complexity features, partially inspired by Horne and Adali (2017), as well as a bag of word n-grams. For tokenization we use the Python Natural Language Toolkit (Bird et al., 2009).

Our features are as follow: *num_sentences* (number of sentences in the document); *avg_sent_word_len* (average word-length of sentences); *avg_sent_char_len* (average character-length of sentences); *var_sent_char_len* (variance of character-length of sentences); *avg_word_len* (average character-length of words); *var_word_len* (variance of character-length of words); *punct_freq* (relative frequency of punctuation characters); *capital_freq* (relative frequency of capital letters); *types_atoms_ratio* (type-token ratio, a measure of vocabulary diversity and richness); and frequency of the k most frequent word n-grams.

Regarding word n-grams, we use $k = 50$ and $n \in [1, 2]$, as we empirically found these values to perform well while maintaining a small feature set. Moreover, we ignore n-grams whose document frequency is greater than 95%, as well as 1-grams from a set of known English stop-words (from *scikit-learn*'s stop-word list), whose frequency we assume to be too high to be distinctive. Text tokens and stop words are *stemmed* using the Porter stemming algorithm (Porter, 1980).

2.2 Hyperparameters

For RF, we use the following hyperparameter values: 100 estimators; minimum samples at leaf = 1; criterion = gini; minimum samples to split = 2.

For GBT, we use the following hyperparameter values: 50 estimators; minimum samples at leaf = 3; loss = exponential; learning rate = 0.3; minimum samples to split = 5; max depth = 8.

For SVM model, we use the following hyperparameter values: penalty parameter C = 0.9; penalty = l2; loss function = squared hinge.

These hyperparameter values are the result of extensive grid searching for each model, selecting the best performing models on 10-fold cross-validated results.

3 Results and Discussion

Table 1 shows the results of the models over 10-fold cross validation (top rows), and on the official test set (bottom rows). Besides our models, we show the performance of the provided baseline as well as the best performing submission to the task (last row). As results on the official test set were hidden during the duration of the task, we used cross-validated results to guide our decision-making in improving the models.

3.1 Feature Analysis

Making use of our choice of classifiers, we are able to interpret and analyze the most important features, as well as trace back the decision path for every document along each of the ensemble's estimators (RF and GBT).

Figures 1 and 2 show measures of feature importance for the RF and GBT models. Figure 1 shows the top features by *mean impurity decrease*

Dataset	*Model*	*Accuracy*	*Precision*	*Recall*	*F1*
10-fold CV on *by-article-training*	GBT	75.9	71.4	59.4	64.6
	RF	76.3	74.6	55.4	63.3
	SVM	72.7	71.3	45.5	55.1
by-article-test (official)	GBT	72.9	78.1	63.7	70.2
	RF	**71.7**	**80.6**	**57.0**	**66.8**
	Baseline	46.2	46.0	44.3	45.1
	Best Team	82.2	87.1	75.5	80.9

Table 1: Models performance: values in the top rows result from 10-fold cross-validation on the *by-article-training* set, and values in the bottom rows report evaluation on the official test set through TIRA (Potthast et al., 2019). RF refers to our task submission, while GBT is our best performing model, submitted after the task's closing.

on a feature's nodes, averaged across the ensemble's estimators/trees and weighted by the proportion of samples reaching those nodes (Breiman, 2001). Similarly, Figure 2 shows the top features by *relative accuracy decrease* (averaged across the ensemble's estimators) as the values of each feature are randomly permuted (Breiman, 2001).

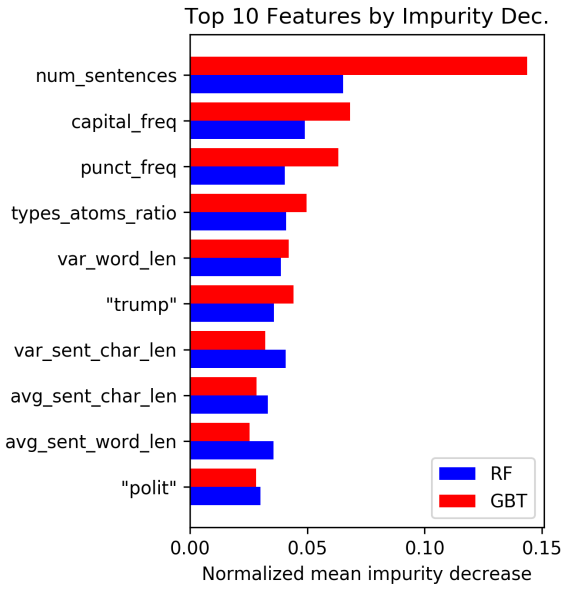

Figure 1: Top features by mean *impurity* decrease, sorted by average value among the two classifiers.

Interesting properties emerge from analyzing feature importance, notably that the *number of sentences* and the *frequency of capital letters* are the most important features on both measures. Moreover, the RF model tends to have a longer-tailed distribution of feature importances, while the GBT model tends to focus on a smaller subset of features for classification.

Interestingly, two 1-grams make it into the top-10 features by impurity decrease: 'trump' and

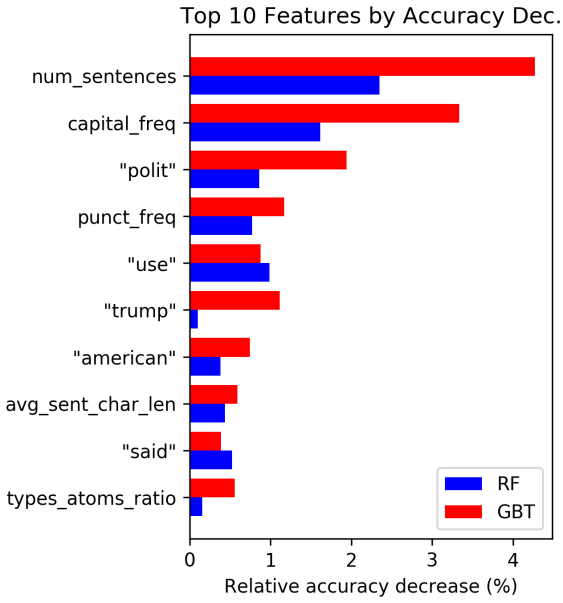

Figure 2: Top features by relative *accuracy* decrease, sorted by average value among the two classifiers.

'polit'. Reliance on n-grams could present a larger problem, as these may refer to entities with a high variance of media attention. For instance, words like 'Hillary' or 'Obama' (which appear in the top-20 features) are probably not seen as often nowadays as they were back in 2016. As such, we are confident in the generalization capacity of the models, as the most discriminative features are mostly style and language-complexity features, which do not suffer from the previously stated biases of n-grams.

3.2 Analysis of Predictions

In order to better understand our model's decision making, we analyze differences in distributions of document features for each predicted class, and compare them with the gold-standard values.

As seen in Table 2, articles predicted as hyperpartisan have a higher number of sentences, but each with lower length than mainstream articles, and with decreased vocabulary diversity (smaller type-token ratio). The frequency of the word 'trump' is also noticeably higher in hyperpartisan articles. There is a good alignment of *predicted* and *gold* articles, when projected onto this feature space.

Feature	Pred. Avg.		Gold Avg.	
	H	M	H	M
num_sentences	39.7	19.4	37.5	20.4
capital_freq	0.046	0.058	0.046	0.058
punct_freq	0.030	0.032	0.030	0.033
type_atoms_ratio	0.54	0.60	0.55	0.59
var_word_len	2.70^2	2.71^2	2.69^2	2.71^2
"trump"	6.13	2.54	5.86	2.64
var_sen_char_len	91.5^2	162^2	93.5^2	162^2
avg_sen_char_len	127	156	129	156
avg_sen_word_len	24.9	29.2	25.1	29.1
"polit"	1.37	0.35	1.34	0.35

Table 2: Average values for articles *predicted* (Pred.) hyperpartisan (H) or mainstream (M), and for their *ground-truth* (Gold), for top-10 features by impurity decrease.

4 Meta-learning Task

After the main task's end, organizers challenged participants to compete on a meta-learning task. This task's dataset consisted of the predictions made by each of the 42 submitted systems on the same test-set articles. Notably, a simple majority vote classifier (with the predictions of all 42 systems as input) achieved accuracy of 88.5%, substantially better than the best performing system's accuracy of 82.2%.

While a voting classifier performed considerably well, we intuitively postulated that the votes of the best-n classifiers (accuracy-wise) would perform better. Figure 3 shows the accuracy of n majority vote classifiers, from the top-*42* systems to the top-*1* system. The best performance is achieved using the top-*12* classifiers. However, in Figure 3, we can observe fluctuations in performance while removing the worst classifiers. This means that combining worst classifiers as we do in this task can yield performance improvements. We conclude that there is no discernible correlation between performance and smaller n. We leave as future work further investigation on what characteristics of the classifiers contribute to the fluctuations of the overall performance.

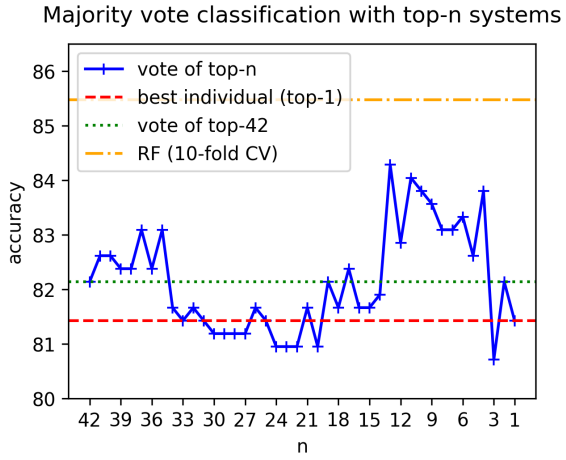

Figure 3: Performance of a majority vote classifier using the top-n best performing systems (by accuracy), on the provided *by-article-meta-training* dataset.

Our final submission for this sub-task consisted of a Random Forest model, whose features were the predictions of all 42 submitted systems, as well as an extra column with the average vote of all systems. See Table 3 for the final performance on the official *by-article-meta-test* dataset.

Model	*A*	*P*	*R*	*F1*
RF	89.9	89.5	90.4	90.0
Majority Vote (42)	88.5	89.2	87.5	88.3
Baseline	52.9	52.5	59.6	55.9

Table 3: Performance on the meta-learning task, evaluated on the *by-article-meta-test* dataset through TIRA.

5 Conclusions

We experimented with several models for hyperpartisan news detection, supplied with a small set of 9 linguistically-inspired features in addition to the 50 most frequent n-grams. Our official submission is a Random Forest model, which achieved an accuracy of 71.7%. On the meta-learning sub-task we achieved an accuracy of 89.9%.

For future work, we intend to further explore differences in writing style between hyperpartisan and mainstream articles, as well as ensembles of individually distinct classifiers, as it seems a promising path towards more accurate hyperpartisan news detection.

Acknowledgments

André Ferreira Cruz is supported by the Calouste Gulbenkian Foundation, under grant number 226338. Gil Rocha is supported by a PhD studentship (with reference SFRH/BD/140125/2018) from Fundação para a Ciência e a Tecnologia (FCT). This research is partially supported by project DARGMINTS (POCI/01/0145/FEDER/031460), funded by FCT.

References

Yaman Akdeniz. 2010. To block or not to block: European approaches to content regulation, and implications for freedom of expression. *Computer Law & Security Review*, 26(3):260–272.

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–36.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Harris Drucker and Corinna Cortes. 1996. Boosting decision trees. In *Advances in neural information processing systems*, pages 479–485.

Jeffrey Gottfried and Elisa Shearer. 2017. Americans online news use is closing in on tv news use. *Pew Research Center*.

Susan C Herring. 2004. Computer-Mediated Discourse Analysis: An Approach to Researching Online Behavior. In S. A. Barab, R. Kling, and J. H. Gray, editors, *Designing for Virtual Communities in the Service of Learning*, pages 338–376. Cambridge University Press, Cambridge.

Tin Kam Ho. 1995. Random decision forests. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282. IEEE.

Benjamin D Horne and Sibel Adali. 2017. This just in: fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In *Eleventh International AAAI Conference on Web and Social Media*.

Wendell Johnson. 1944. Studies in language behavior: A program of research. *Psychological Monographs*, 56(2):1–15.

Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney, Benno Stein, and Martin Potthast. 2019. SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval 2019)*. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Martin F Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.

Martin Potthast, Tim Gollub, Matti Wiegmann, and Benno Stein. 2019. TIRA Integrated Research Architecture. In Nicola Ferro and Carol Peters, editors, *Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF*. Springer.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. A Stylometric Inquiry into Hyperpartisan and Fake News. In *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 231–240. Association for Computational Linguistics.

Craig Silverman, Lauren Strapagiel, Hamza Shaban, Ellie Hall, and Jeremy Singer-Vine. 2016. Hyperpartisan facebook pages are publishing false and misleading information at an alarming rate. *Buzzfeed News*.

Rui Sousa-Silva, Gustavo Laboreiro, Luís Sarmento, Tim Grant, Eugénio Oliveira, and Belinda Maia. 2011. twazn me!!! ;(' Automatic Authorship Analysis of Micro-Blogging Messages. In *Lecture Notes in Computer Science 6716 Springer 2011*, volume Natural La, pages 161–168, Berlin and Heidelberg. Springer – Verlag.

Rui Sousa-Silva, Luís Sarmento, Tim Grant, Eugénio C Oliveira, and Belinda Maia. 2010. Comparing Sentence-Level Features for Authorship Analysis in Portuguese. In *Computational Processing of the Portuguese Language*, pages 51–54.

Efstathios Stamatatos. 2009. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556.