SemEval-2019 Task 4: Hyperpartisan News Detection

Johannes Kiesel[1] **Maria Mestre**[2,3] **Rishabh Shukla**[2] **Emmanuel Vincent**[2]
Payam Adineh[1] **David Corney**[4] **Benno Stein**[1] **Martin Potthast**[5]

[1] Bauhaus-Universität Weimar
<first>.<last>@uni-weimar.de

[2] Factmata Ltd.
<first>.<last>@factmata.com

[3] mariarmestre@gmail.com
[4] dpacorney@gmail.com

[5] Leipzig University
martin.potthast@uni-leipzig.de

Abstract

Hyperpartisan news is news that takes an extreme left-wing or right-wing standpoint. If one is able to reliably compute this meta information, news articles may be automatically tagged, this way encouraging or discouraging readers to consume the text. It is an open question how successfully hyperpartisan news detection can be automated, and the goal of this SemEval task was to shed light on the state of the art. We developed new resources for this purpose, including a manually labeled dataset with 1,273 articles, and a second dataset with 754,000 articles, labeled via distant supervision. The interest of the research community in our task exceeded all our expectations: The datasets were downloaded about 1,000 times, 322 teams registered, of which 184 configured a virtual machine on our shared task cloud service TIRA, of which in turn 42 teams submitted a valid run. The best team achieved an accuracy of 0.822 on a balanced sample (yes : no hyperpartisan) drawn from the manually tagged corpus; an ensemble of the submitted systems increased the accuracy by 0.048.

1 Introduction

Yellow journalism has established itself in social media, nowadays often linked to phenomena like clickbait, fake news, and hyperpartisan news. Clickbait has been its first "success story" (Potthast et al., 2016): When the viral spreading of pieces of information was first observed in social networks, some investigated how to manufacture such events for profit. Unlike for "natural" viral content, however, readers had to be directed to a web page containing the to-be-spread information alongside paid-for advertising, so that only teasers and not the information itself could be shared. Then, to maximize their virality, data-driven optimization revealed that teaser messages which induce curiosity, or any other kind of strong emotion, spread best. The many forms of such teasers that have emerged since are collectively called clickbait. New publishing houses arose around viral content, which brought clickbait into the mainstream. Traditional news publishers, struggling for their share of the attention market that is a social network, adopted clickbait into their toolbox, too, despite its violation of journalistic codes of ethics.

The content spread using clickbait used to be mostly harmless trivia—entertainment and distraction to some, spam to others—, but in the wake of the 2016 United States presidential election, "fake news" came to widespread public attention. While certainly not a new phenomenon in yellow journalism, its viral success on social media was a surprise to many. Part of this success was then attributed to so-called hyperpartisan news publishers (Bhatt et al., 2018), which report strongly in favor of one political position and in fierce disagreement with its opponents. Clinging to hyperpartisanship often entails stretching the truth, if not breaking it with fake news, whose highly emotional content makes them spread exceptionally fast, like clickbait.

Given the hype surrounding fake news, activists, industry, and research are now paying a lot of attention to mitigating the problem, such as trying to check facts in news items. Clickbait and hyperpartisan news, however, have been less studied. In previous work, we sought to help close this gap from both ends: for clickbait detection (Potthast et al., 2016), part of our group created a large-scale evaluation dataset (Potthast et al., 2018b) and set up an ongoing competition for the best detection approach (Potthast et al., 2018a). For hyperpartisan news detection (Potthast et al., 2018c), we teamed up to follow a similar approach that led to the Hyperpartisan News Detection task at SemEval-2019. This paper reports on the results of this task.

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 829–839
Minneapolis, Minnesota, USA, June 6–7, 2019. ©2019 Association for Computational Linguistics

2 Task Definition

We define hyperpartisan news detection as follows:

> Given the text and markup of an online news article, decide whether the article is hyperpartisan or not.

Hyperpartisan articles mimic the form of regular news articles, but are one-sided in the sense that opposing views are either ignored or fiercely attacked. We deliberately disregard the distinction between left and right, since previous work has found that, in hyperpartisan form, both are more similar to each other in terms of style than either are to the mainstream (Potthast et al., 2018c). The challenge of this task is to unveil the mimicking and to detect the hyperpartisan language, which may be distinguishable from regular news at the levels of style, syntax, semantics, and pragmatics.

3 Data

Our focus is on news articles published online, and we provide two datasets with this task. One has 1,273 articles, each labeled manually, while the second, larger dataset of 754,000 articles is labeled in a semi-automated manner via distant supervision at the publisher level. These datasets are further split into public and private sets. We released the public set for the model training, tuning, and evaluation,[1] while the unreleased private set is used to enable blind, cloud-based evaluation.

As online news articles are published mainly in the HTML format, both datasets use a unified HTML-like format (see Figure 1). We restricted the markup for the article content to paragraphs (<p>), links (<a>), and quotes (<q>). We distinguished *internal* links to the other pages of the same domain, from which we removed the href-attribute value to avoid classifiers fitting to them; and links to *external* domains, for which we kept the attribute. An XML schema that exactly specifies the format is distributed along the datasets.

3.1 Dataset Annotated By Article

We gathered a crowdsourced dataset of 1,273 articles, each labeled manually by 3 annotators (Vincent and Mestre, 2018). These articles were published by active hyperpartisan and mainstream websites and were all assured to contain political news. Annotators were asked to rate each article's bias on the following 5-point Likert scale:

1. No hyperpartisan content
2. Mostly unbiased, non-hyperpartisan content
3. Not Sure
4. Fair amount of hyperpartisan content
5. Extreme hyperpartisan content

We removed all articles from the dataset with low agreement score and the aggregated rating of "not sure" (see Vincent and Mestre for more details). We then binarized the labels to hyperpartisan (average rating of 4 or 5) and not (average rating of 1 or 2). The final by-article set achieved an inter-annotator agreement of 0.5 Krippendorff's alpha. Of the remaining 1,273 articles, 645 were published as a training dataset, whereas the other 628 (50% hyperpartisan and 50% not) were kept private for the evaluation. To ensure that classifiers could not profit from overfitting to publisher style, we made sure there was no overlap between the publishers of the articles between these two sets.

3.2 Dataset Annotated By Publisher

To allow for methods that require huge amounts of training data, we compiled a dataset of 754,000 articles, each labeled as per the bias of their respective publisher. To create this dataset, we cross-checked two publicly available news publisher bias lists compiled by media professionals from BuzzFeed news[2] and Media Bias Fact Check.[3] The former was created by BuzzFeed journalists as a basis for a news article, whereas the latter is Media Bias Fact Check's main product. While both lists contain several hundred news publishers, they disagree only for nine, which we removed from our dataset.

We then crawled, archived, and post-processed the articles available on the publishers' web sites and Facebook feeds. We archived all articles using a specialized tool (Kiesel et al., 2018) that removes pop-overs and similar things preventing the article content from being loaded. After filtering out publishers that did not mainly publish political articles or had no political section to which we could restrict our crawl, we were left with 383 publishers. For each of the publishers' web sites we wrote a content-wrapper to extract the article content and relevant meta data from the HTML DOM. We then removed all articles that were too short to contain news,[4] that are not written in English,

[1] https://doi.org/10.5281/zenodo.1489920

[2] https://github.com/BuzzFeedNews/2017-08-partisan-sites-and-facebook-pages

[3] https://mediabiasfactcheck.com

[4] Based on manual inspection of a hundred short articles, we set the threshold to 40 words.

```
<article id="0182515" published-at="2007-01-22" title="They’re crumbling">
<p>What a pleasant surprise to see Jacques Leslie, a journalist and real expert on
dams, with a long <a href="http://www.nytimes.com/2007/01/22/opinion/22leslie.2.html
?ex=1327122000&amp;amp;en=42caf99f05e4cba8&amp;amp;ei=5090&amp;amp;partner=
rssuserland&amp;amp;emc=rss" type="external">op-ed</a> on the hallowed pages of the
New York Times.  Leslie, author of <a href="" type="internal">Deep Water:  The Epic
Struggle Over Dams, Displaced People and the Environment</a>, highlights the threat
posed by poorly maintained and increasingly failing dams around the country:</p>
<p>Unlike, say, waterways and sanitation plants, a majority of dams - 56 percent of
those inventoried - are privately owned, which is one reason dams are among the
country’s most dangerous structures.  Many private owners can’t afford to repair
aging dams; some owners go so far as to resist paying by tying up official repair
demands in court or campaigning to weaken state dam safety laws.</p>
<p>Kinda makes you want to find out what is upstream.</p> </article>
```

Figure 1: Example of a non-hyperpartisan article in our dataset. An archived version of the original article is available at https://web.archive.org/web/20121006194050/https://grist.org/article/remember-the-dams/.

or that contain obvious encoding errors. The final dataset consisted of 754,000 articles, split into a public training set (600,000 articles), a public validation set (150,000 articles) and a non-public test set (4,000 articles). Like for the by-article dataset, we ensured that there is no overlap of publishers between the sets. Each set consists of 50% articles from non-hyperpartisan publishers and 50% articles from hyperpartisan publishers, the latter again being 50% from left-wing and 50% from right-wing publishers.

4 Fairness and Reproducibility

In this shared task, we asked participants to submit their software instead of just its run output. The submissions were executed at our site on the test data, enabling us to keep the test data entirely secret. This has two important advantages over traditional shared task setups: first, software submission gives rise to blind evaluation; and second, it maximizes the replicability and the reproducibility of each participant's approach. To facilitate software submission and to render it feasible in terms of work overhead and flexibility for both participants and organizers, we employ the TIRA Integrated Research Architecture (Potthast et al., 2019).

A shortcoming of traditional shared task setups is that typically the test data are shared with participants, albeit without ground truth. Although participants in shared tasks generally exercise integrity and do not analyze the test data other than running their software on it, we have experienced cases to the contrary. Such problems particularly arise in shared tasks where the stakes are higher than usual; when monetary incentives are offered or winning results in high visibility. A partial workaround is to share the test data only very close to the final submission deadline, minimizing analysis opportunities. But if sharing the test data is impossible for reasons of sensibility and proprietariness, or because the ground truth can be easily reverse-engineered, a traditional shared task cannot be held.

Another shortcoming of traditional shared tasks (and many computer science publications in general) is their lack of reproducibility. Although sharing the software underlying experiments as well as the trained models is easy, and although it would greatly aid reproducibility, this is still rare. Typically, all that remains after a shared task are the papers and datasets published. Given that shared tasks often establish a benchmark for the task in question, acting normative for future evaluations, this outcome is far from optimal and comparably wasteful. All of the above can be significantly improved upon by asking participants not to submit their software's run output, but the software itself. However, this entails a significant work overhead for organizers, especially for larger tasks.

In order to mitigate the work overhead, we employ TIRA. In a nutshell, TIRA implements evaluation as a service in the form of a cloud-based evaluation platform. Participants deploy their software into virtual machines hosted at TIRA's cloud, and then remotely control the machines and the software within, executing it on the test data. The test data are available only within the cloud, and made accessible on demand so that participants cannot access it directly. At execution time, the virtual machine is disconnected from the internet, copied, and only the copy gets access to the test data. Once the automatically executed software terminates, its run output is saved and the virtual machine copy is destroyed so as to prevent data leaks. This way, all submitted pieces of software can be archived in working condition, and be re-evaluated at a later time, even on new datasets.

5 Participating Systems

This task attracted a very diverse and interesting set of solutions from the participating teams. The teams employed very different sets of features, a wide variety of classifiers, and also employed the large by-publisher dataset in different ways. Around half of the submissions used hand-crafted features. In the following, we give an overview of the submitted approaches. For a more readable and condensed form, we only use the team names here, which were chosen from fictional journalistic characters or entities (see Table 1 for references).

5.1 Features

The teams that participated in this task employed a variety of features, including standard word n-grams (also unigrams, i.e., bag-of-words), word embeddings, stylometric features, HTML features like the target of hyperlinks, and a meta data feature in the form of the publication date.

N-Grams Most teams that used hand-crafted features also included word n-grams: Pioquinto Manterola and Tintin used them as their only features. Character and part-of-speech n-grams were, for example, used by Paparazzo.

Word embeddings Many teams integrated word embeddings into their approach. Frequently used were Word2Vec, fastText, and GloVe. Noticeably, Tom Jumbo Grumbo relied exclusively on them. Bertha von Suttner relied on ELMo embeddings (Peters et al., 2018), which have the advantage of modeling polysemy. Where the aforementioned word embeddings all rely on neural networks, Doris Martin employed a document representation based on word clusters as part of their approach.

BERT (Devlin et al., 2018), which jointly conditions on both left and right context in all layers, is a rather new technique that was used by several teams. Peter Parker directly applied a freely available pre-trained BERT model to the task, whereas Howard Beale and Clint Buchanan trained their own BERT models on the by-publisher dataset and then performed fine-tuning on the by-article dataset. Despite the fine-tuning, Howard Beale reported overfitting issues for this strategy. Going one step further, Jack Ryder and Yeon Zi integrated BERT in their neural network architectures.

Stylometry Many teams used stylometric features including punctuation and article structure (Steve Martin, Spider Jerusalem, Fernando Pessa, Ned Leeds, Carl Kolchak, Orwellian Times), readability scores (Ned Leeds, Pistachon, Steve Martin, Orwellian Times, D X Beaumont), or psycholinguistic lexicons (Ned Leeds, Spider Jerusalem, Steve Martin, Pistachon). Borat Sagdiyev employed a self-compiled list of trigger words that contains mostly profanities. They noticed that such words are used more often in hyperpartisan articles.

Emotionality Several teams used sentiment and emotion features, either based on libraries (Borat Sagdiyev, Steve Martin, Carl Kolchak) or lexicons (Spider Jerusalem, D X Beaumont). Notably, Kermit the Frog uses sentiment detection only. Vernon Fenwick and D X Beaumont used subjectivity and polarity metrics as features.

Named entities Borat Sagdiyev used named entity types as features. In preliminary tests only the type of “nationalities or religious and political groups” was found to be predictive.

Quotations A few teams treated quotations separately. Whereas Spider Jerusalem and Borat Sagdiyev created separate features from quotations, the Ankh Morpork Times filtered them out for not necessarily representing the views of the author.

Hyperlinks Only few teams considered hyperlinks. Both Borat Sagdiyev and Steve Martin used external lists of partisan web pages to count how often an article links to partisan and non-partisan pages. They assume that articles tend to link other articles on the same side of the political spectrum.

Publication date Based on the conjecture that months around American elections could see more hyperpartisan activity, Borat Sagdiyev used the publication month and year as separate features.

5.2 Classifiers

While many different classifiers were used overall, neural networks were the most frequent, which mirrors the current trend in text classification.

The most popular type of neural networks among the participants were convolutional ones (CNNs), which employ convolving filters over neighboring words. Many teams cited the architecture by Kim (2014). Xenophilius Lovegood added a second layer to their CNN in order to encode more information about the articles, using both available and custom-learned embeddings. While Pioquinto Manterola experimented with a CNN, it suffered from overfitting and was thus not used for the final submission. Peter Brinkmann built a submission using available embeddings. Brenda Starr combined a CNN with a sentence-level bidirectional recurrent neural network and an attention mechanism to a complex architecture. A similar approach was employed by the Ankh Morpork Times. An ensemble of three CNN-based models was used by Bertha von Suttner. Steve Martin used a character bigram CNN as part of their approach.

Next to CNNs, long short term memory networks (LSTM) were employed by Kit Kittredge and Miles Clarkson. The latter extended the network with an attention model. Moreover, Joseph Rouletabille used the hierarchical attention network of Yang et al. (2016).

Besides neural networks, a wide variety of classifiers were used. A few teams opted for SVMs (e.g., the Orwellian Times), others for random forests (e.g., Fernando Pessa), linear models (e.g., Pistachon), the Naive Bayes model (e.g., Carl Kolchak), XGBOOST (Clark Kent), Maxent (Doris Martin), and rule-based models (Harry Friberg). Morbo used ULMFit (Howard and Ruder, 2018) to adapt a language model pre-trained on Wikipedia articles to the articles and classes of this task.

5.3 Usage of the By-publisher Dataset

The submitted systems can also be distinguished by whether and how they used the large, distantly-supervised by-publisher dataset. Though much larger than the by-article set, its labels are noisy, whereas the opposite holds for the by-article dataset. One of the key challenges faced by many teams was how to train a powerful expressive model on the smaller dataset without overfitting. Most teams made use of the larger dataset in some form or another. A challenge faced by some of the teams was that the test split of the by-article dataset was balanced between classes, whereas the corresponding training dataset was not.

Several systems trained the whole or part of their system on the by-publisher dataset. Some extracted features like n-grams (e.g., Sally Smedley), word clusters (Doris Martin), or neural network word embeddings (e.g., Clint Buchanan). Others used the larger dataset to perform hyperparameter search (e.g., Miles Clarkson). Many teams trained their models using the by-publisher dataset only (Pistachon, Joseph Rouletabille, Xenophilius Lovegood, Peter Brinkmann, and Kit Kittredge).

To reduce the noise in the distantly-supervised data, some teams used only a subset of it. Yeon Zi, Borat Sagdiyev and the Anhk Morpork Times fitted a model on the by-article dataset and ran it on the by-publisher one: the articles of the by-publisher dataset that were misclassified by this model, were presumed to be noisy and filtered out.

6 Results

A total of 42 teams completed the task, representing more than twenty countries between them, including India, China, the USA, Japan, Vietnam, and many European countries. Table 1 shows the accuracy, precision, recall, and F_1 score for each team, sorted by accuracy. This task used accuracy as the main metric to represent a filtering scenario. The accuracy scores ranged from 0.462 up to 0.822.

The results show a range of trade-offs between precision and recall and the resulting F_1 scores. The highest F_1 was 0.821 with a precision of 0.815 and a recall of 0.828; the highest precision was 0.883 with a recall of 0.672 (F_1: 0.763); and the highest recall was 0.971 with a relatively low precision of 0.542 (F_1: 0.696).

6.1 Methods Used by the Top Teams

While the winning team, Bertha von Suttner, used deep learning (sentence-level embeddings and a convolutional neural network) the second-placed team, Vernon Fenwick, took a different approach and combined sentence embeddings with more domain-specific features and a linear model. Out of the top five teams, only two used “pure” deep learning models of neural networks without any domain-specific, hand-crafted features, showing no single method has a clear advantage over others.

Bertha von Suttner used a model based on ELMo embeddings (Peters et al., 2018) and trained on the by-article dataset. After minimal preprocessing, a pre-trained ELMo was applied onto each token of each sentence, and then averaged, to obtain average sentence embeddings. The sentence embeddings were later passed through a CNN, batch-normalized, followed by a dense layer and a sigmoid function to obtain the final probabilities. The final model was an ensemble of the 3 best-performing models of a 10-fold cross-validation. The authors tried to include the by-publisher dataset, but found in their preliminary tests no approach to profit from the large data.

Submission			By-article dataset					By-publisher dataset				
Team name	Authors	Code	Rank	Acc.	Prec.	Recall	F_1	Rank	Acc.	Prec.	Recall	F_1
Bertha von Suttner	Jiang et al.	🔗	1	**0.822**	0.871	0.755	0.809	8	0.643	0.616	0.762	0.681
Vernon Fenwick	Srivastava et al.		2	0.820	0.815	0.828	**0.821**					
Sally Smedley	Hanawa et al.		3	0.809	0.823	0.787	0.805	11	0.625	0.640	0.571	0.603
Tom Jumbo Grumbo	Yeh et al.	🔗	4	0.806	0.858	0.732	0.790	13	0.619	0.592	0.762	0.667
Dick Preston	Isbister and Johansson		5	0.803	0.793	0.818	0.806	27	0.514	0.520	0.352	0.420
Borat Sagdiyev	Palić et al.		6	0.791	**0.883**	0.672	0.763	19	0.592	0.644	0.412	0.502
Morbo	Isbister and Johansson		7	0.790	0.772	0.822	0.796	16	0.601	0.587	0.679	0.630
Howard Beale	Mutlu et al.		8	0.783	0.837	0.704	0.765	9	0.641	0.606	0.806	0.692
Ned Leeds	Stevanoski and Gievska		9	0.775	0.865	0.653	0.744	22	0.573	0.546	0.857	0.667
Clint Buchanan	Drissi et al.	🔗	10	0.771	0.832	0.678	0.747					
Yeon Zi	Lee et al.		11	0.758	0.744	0.787	0.765	5	0.663	0.635	0.766	0.694
Tony Vincenzo	Staykovski		12	0.750	0.764	0.723	0.743					
Paparazzo	Nguyen et al.	🔗	13	0.747	0.754	0.732	0.743	24	0.530	0.530	0.541	0.535
Steve Martin	Joo and Hwang		14	0.745	0.853	0.592	0.699	18	0.597	0.625	0.483	0.545
Eddie Brock	Šajatović et al.		15	0.744	0.782	0.675	0.725	10	0.631	0.681	0.491	0.571
Ankh Morpork Times	Almendros et al.		16	0.742	0.811	0.631	0.710	21	0.588	0.646	0.389	0.486
Spider Jerusalem	Alabdulkarim and Alhindi	🔗	17	0.742	0.814	0.627	0.709					
Carl Kolchak	Chen et al.		18	0.739	0.729	0.761	0.745					
Doris Martin	Agerri	🔗	19	0.737	0.754	0.704	0.728					
Pistachon	Saleh et al.		20	0.729	0.724	0.742	0.733	15	0.608	0.638	0.499	0.560
Joseph Rouletabille	Moreno et al.		21	0.725	0.788	0.615	0.691	2	0.680	0.640	0.827	**0.721**
Fernando Pessa	Cruz et al.	🔗	22	0.717	0.806	0.570	0.668	17	0.600	0.585	0.681	0.630
Pioquinto Manterola	Sengupta and Pedersen	🔗	23	0.704	0.741	0.627	0.679					
Miles Clarkson	Zhang et al.		24	0.683	0.745	0.557	0.638	6	0.652	0.612	0.832	0.705
Xenophilius Lovegood	Zehe et al.		25	0.675	0.619	0.914	0.738	4	0.663	0.632	0.781	0.699
Orwellian Times	Knauth		26	0.672	0.654	0.729	0.690	23	0.537	0.530	0.658	0.587
Tintin	Bestgen		27	0.656	0.642	0.707	0.673	1	**0.706**	0.742	0.632	0.683
D X Beaumont	Amason et al.		28	0.653	0.597	0.939	0.730					
Jack Ryder	Shaprin et al.		29	0.646	0.646	0.646	0.646	7	0.645	0.600	**0.869**	0.710
Kermit the Frog	Anthonio and Kloppenburg		30	0.621	0.582	0.860	0.694	20	0.589	0.575	0.681	0.623
Billy Batson	Kreutz et al.		31	0.615	0.568	0.962	0.714					
Peter Brinkmann	Färber et al.	🔗	32	0.602	0.560	0.955	0.706	28	0.497	0.496	0.344	0.406
Anson Bryson	Stiff and Medero		33	0.592	0.720	0.303	0.426					
Sarah Jane Smith	Chakravartula et al.		34	0.591	0.554	0.933	0.695	14	0.612	0.586	0.765	0.664
Kit Kittredge	Cramerus and Scheffler		35	0.578	0.547	0.908	0.683					
Brenda Starr	Papadopoulou et al.		36	0.575	0.542	**0.971**	0.696	3	0.664	0.627	0.807	0.706
Harry Friberg	Afsarmanesh et al.		37	0.565	0.537	0.949	0.686					
Robin Scherbatsky	Marx and Akut		38	0.551	0.542	0.662	0.596	25	0.524	**0.822**	0.062	0.116
Clark Kent	Gupta et al.	🔗	39	0.548	0.683	0.178	0.283	26	0.519	0.565	0.170	0.261
Murphy Brown	Sen and Jiang		40	0.529	0.518	0.822	0.635	12	0.623	0.615	0.659	0.636
Peter Parker	Ning et al.		41	0.503	0.502	0.771	0.608					
John King	Bansal et al.		42	0.462	0.460	0.443	0.451					

Table 1: For each team and dataset, the performance of the submission that reached the highest accuracy is shown. If a team published their code, the 🔗 links to the respective repository. We forked all repositories for archival.[6]

The second and third best teams used linear models as their main predictor and embeddings as features, training on the by-article dataset only. Vernon Fenwick extracted sentence embeddings with the Universal Sentence Encoder (USE) (Cer et al., 2018), while Sally Smedley used BERT to generate contextual embeddings. Both teams also employed hand-crafted, domain-specific features. Vernon Fenwick extracted article-level and sentence-level polarity, bias, and subjectivity, among others, while Sally Smedley used the by-publisher dataset to extract key discriminative phrases, which they later looked up in the training data.

6.2 Overall Insights

The results reveal several insights into the suitability of different features and approaches for the task of hyperpartisan news detection.

Word-embeddings have been reported to be a very efficient feature by many teams. Tom Jumbo Grumbo achieved an accuracy of 0.806 with GloVe embeddings and a classifier trained on the by-article dataset. The application of a pre-trained BERT model by Peter Parker performed very poorly (acc. 0.503). However, the same BERT embeddings were used for great effect by Sally Smedley, using techniques like word-dropout and informative phrase identification (acc. 0.809).

[6] https://github.com/hyperpartisan-news-challenge

Also standard word n-grams were found to be suitable for the task, though not as strong as embeddings. While n-grams where used in several well-performing approaches, Pioquinto Manterola reached an accuracy of 0.704 with unigrams alone.

Several teams reported an increase in accuracy through sentiment or similar features (e.g., Borat Sagdiyev). Kermit the Frog used sentiment detection alone to reach an accuracy of 0.621.

Besides textual features, a few teams also analyzed HTML and article meta-features. Borat Sagdiyev performed a detailed analysis in this regard, which helped them to achieve the highest precision of all teams. For example, they found that both the publication date and the number of links to known hyperpartisan pages could each improve the overall accuracy by about 0.01 to 0.02.

Of the top teams, only Sally Smedley used the by-publisher dataset, and only to select n-grams. Based on the reports of several teams, the utilization of this dataset thus seems more difficult than we expected. We conjecture that this is due to the mis-classification of what should be the most informative articles: non-hyperpartisan articles from mainly hyperpartisan publishers, and hyperpartisan articles from non-hyperpartisan publishers. These articles are especially suited to distinguish features that identify hyperpartisanship from features that identify publisher style. While we assumed that the advantages of big data would outweigh this drawback, the results suggest that it might be more worthwhile to put effort in larger datasets where each article is annotated separately. Still, some teams managed to use the by-publisher dataset as a large dataset of in-domain texts. For example, Clint Buchanan reported that pre-training embeddings on the by-publisher dataset increased the accuracy of their system on the by-article dataset.

Moreover, the ranking of teams for the two test datasets is quite different. Bertha von Suttner, who ranked first for by-article, reached only rank eight for the by-publisher dataset. Conversely, Tintin, who optimized for by-publisher, ranked first there but only 27th for the by-article dataset. This discrepancy highlights the unexpected large differences between the datasets.

7 Meta-Classification Task

Inspired by successes of meta classifiers in past SemEval tasks (e.g., Hagen et al. (2015)), we enabled and encouraged participants to devise meta classifiers that learn from the classifications of the submitted approaches. For this meta-classification task, we split the test datasets further into new training (66%) and test sets (33%). We again made sure that there are an equal amount of non-hyperpartisan and hyperpartisan articles, as well as an equal share of left-wing and right-wing articles within the hyperpartisan sets. Furthermore, we again assured that no publisher had articles in both the training and the test sets. An instance in these datasets corresponds to the classifications (hyperpartisan or not) of the best-performing software of each team (42 classifications for the by-article dataset and 30 for the by-publisher one) of one article from the original test data.

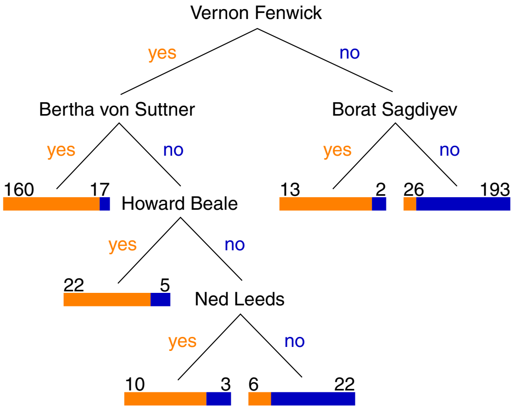

Figure 2: Meta-classification decision tree J48-M10 learned on the predictions of the submitted systems (hyperpartisan: yes or no; by-article dataset). The numbers show the training class-distribution at the leafs.

We provide two simple classification systems for baselines, majority voting and an out-of-the-box decision tree, which both outperform the best single submitted software and which were both outperformed by the meta-classifiers submitted. Majority voting refers to a system that outputs the classification (hyperpartisan or not) that the most base classifiers selected. As it does not learn a decision boundary, it is—strictly speaking—not a meta classifier. For the decision tree, we used the J48 implementation of WEKA (Frank et al., 2016). We tested two variants: standard settings (*J48-M2*) and restricting leaf nodes to contain at least 10 articles (*J48-M10*) to force a simpler decision tree. Simpler trees often generalize better to unseen data.

Figure 2 shows the J48-M10 tree for the by-article dataset. For every leaf of the tree, more than 75% of the corresponding training articles are from the same class. This shows that even with as few as 5 decision nodes, the training set

Team or system name	Acc.	Prec.	Recall	F_1
Fernando Pessa	**0.899**	0.895	**0.904**	**0.900**
Spider Jerusalem	**0.899**	0.903	0.894	0.899
Majority Vote	0.885	0.892	0.875	0.883
J48-M10	0.880	**0.916**	0.837	0.874
J48-M2	0.856	0.863	0.846	0.854
Bertha von Suttner alone	0.851	0.901	0.788	0.841

Table 2: Accuracy, precision, recall, and F_1-measure for the by-article meta learning test dataset.

could be fitted reasonably well. The meta classifier was thus able to use the submitted systems as predictive and distinct features, which shows that some submitted systems performed well on some articles where other systems did not and vice versa. Even more, the 5 systems employed by the meta-classifier are all within the top 10 systems of the task, which shows that there is considerable variation even among the top performers. This is reasonable, given the variety of approaches used.

In addition to our approaches, two teams submitted their own classifiers in the short time span they had. Fernando Pessa used a random forest classifier trained on the single predictions as well as the average vote. Spider Jerusalem used a weighted majority voting algorithm, where they weighted each single prediction by the precision of the respective classifier on the training set.

Table 2 shows the performance of the approaches on the meta learning test dataset. Note that the best single system, Bertha von Suttner, reaches an increased accuracy of 0.851 on the meta learning test set. This is due to variations in the small dataset. Still, all ensemble approaches reach a higher accuracy. The majority voting approach reaches an accuracy of 0.885, and thus outperforms the J48 classifiers. This is somewhat surprising, but shows that there is a lot to gain by integrating also the systems that performed less well—team Fernando Pessa came to a similar insight in their paper (Cruz et al., 2019). The approaches of the two participants performed very similar, despite their methodological differences, and outperformed the majority vote. They managed to achieve an accuracy 0.048 points above Bertha von Suttner and therefore a considerable increase in performance.

We also repeated the experiments for the by-publisher dataset, but could not produce decisive results there, yet. We assume that this is due to most teams focusing on the other dataset and both datasets being more different than expected.

8 Conclusion

This paper reports on the setup, participation, results, and insights gained from the first task in hyperpartisan news detection, hosted as Task 4 at SemEval-2019. We detailed the construction of both a manually annotated dataset of 1,273 articles as well as a large dataset of 754,000 articles, compiled using distant supervision. Moreover, it provides a systematic overview of the 34 papers submitted by the participants, insights gathered from single teams, by comparing their approaches, and by an ad-hoc meta classification.

Through the use of TIRA (Potthast et al., 2019), we were able to establish a blind evaluation setup, so that future approaches can be compared on same grounds. For this, we continue to accept new approaches in ongoing submissions.[7] Moreover, through the use of TIRA we can directly evaluate the submitted approaches on new datasets for hyperpartisan news detection, provided they are formatted like the datasets presented here.

Very promising results were achieved during the task, with accuracy values above 80% on a balanced test set—and even up to 90% using meta classification on all submissions. Like in many other NLP tasks, word embeddings could be used to great effect, but hand-crafted features also performed well. The differences between the two employed datasets were larger than anticipated, which suggests a focus on by-article annotations in the future. A larger dataset of this kind will probably assist in improving the accuracy of future models even beyond the already very good level.

It thus seems that hyperpartisan news detection is already sufficiently developed to take the next step and demand human-understandable explanations from the approaches. The most obvious use cases of hyperpartisan news detectors are for filtering articles, which always requires a careful handling to avoid unwarranted censorship. Especially in the current political climate, it therefore seems necessary that hyperpartisanship detectors not only reach a high accuracy, but also reveal their reasoning.

Acknowledgements

Our thanks go out to all participating teams; your contributions made this task a success. We hope we have been able to do your work justice, and are looking forward to doing so in the future. Our special thanks go out to the SemEval organizers for providing perfect organizational support.

[7] https://webis.de/events/semeval-19/

References

Nazanin Afsarmanesh, Jussi Karlgren, Peter Sumbler, and Nina Viereckel. 2019. Team Harry Friberg at SemEval-2019 Task 4: Identifying Hyperpartisan News through Editorially Defined Metatopics. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Rodrigo Agerri. 2019. Doris Martin at SemEval-2019 Task 4: Hyperpartisan News Detection with Generic Semi-supervised Featuresl. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Amal Alabdulkarim and Tariq Alhindi. 2019. Spider-Jerusalem at SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Carla Perez Almendros, Luis Espinosa Anke, and Steven Schockaert. 2019. Cardiff University at SemEval-2019 Task 4: Linguistic Features for Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Evan Amason, Jake Palanker, Mary Clare Shen, and Julie Medero. 2019. Harvey Mudd College at SemEval-2019 Task 4: The D.X. Beaumont Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Talita Anthonio and Lennart Kloppenburg. 2019. Team Kermit-the-frog at SemEval-2019 Task 4: Bias Detection Through Sentiment Analysis and Simple Linguistic Features. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Yves Bestgen. 2019. Tintin at SemEval-2019 Task 4: Detecting Hyperpartisan News Article with only Simple Tokens. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Shweta Bhatt, Sagar Joglekar, Shehar Bano, and Nishanth Sastry. 2018. Illuminating the ecosystem of partisan websites. In *Proceedings of the 27th International Conference on World Wide Web Companion*, WWW '18 Companion. International World Wide Web Conferences Steering Committee.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder.

Nikhil Chakravartula, Vijayasaradhi Indurthi, and Bakhtiyar Syed. 2019. Fermi at SemEval-2019 Task 4: The sarah-jane-smith Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Celena Chen, Celine Park, Jason Dwyer, and Julie Medero. 2019. Harvey Mudd College at SemEval-2019 Task 4: The Carl Kolchak Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Rebekah Cramerus and Tatjana Scheffler. 2019. Team Kit Kittredge at SemEval-2019 Task 4. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

André Cruz, Gil Rocha, Rui Sousa-Silva, and Henrique Lopes Cardoso. 2019. Team Fernando-Pessa at SemEval-2019 Task 4: Back to Basics in Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Mehdi Drissi, Pedro Sandoval Segura, Vivaswat Ojha, and Julie Medero. 2019. Harvey Mudd College at SemEval-2019 Task 4: The Clint Buchanan Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Michael Färber, Agon Qurdina, and Lule Ahmedi. 2019. Team Peter Brinkmann at SemEval-2019 Task 4: Detecting Biased News Articles Using Convolutional Neural Networks. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Eibe Frank, Mark A. Hall, and Ian H. Witten. 2016. *Data Mining: Practical Machine Learning Tools and Techniques*, 4th edition, chapter The WEKA Workbench. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Viresh Gupta, Baani Leen Kaur Jolly, Ramneek Kaur, and Tanmoy Chakraborty. 2019. Clark Kent at SemEval-2019 Task 4: Stylometric Insights into Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Matthias Hagen, Martin Potthast, Michel Büchner, and Benno Stein. 2015. Webis: An Ensemble for Twitter Sentiment Detection. In *9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 582–589. Association for Computational Linguistics.

Kazuaki Hanawa, Shota Sasaki, Hiroki Ouchi, Jun Suzuki, and Kentaro Inui. 2019. The Sally Smedley Hyperpartisan News Detector at SemEval-2019 Task 4. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. *CoRR*, abs/1801.06146.

Tim Isbister and Fredrik Johansson. 2019. Dick-Preston and Morbo at SemEval-2019 Task 4: Transfer Learning for Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Ye Jiang, Johann Petrak, Xingyi Song, Kalina Bontcheva, and Diana Maynard. 2019. Team Bertha von Suttner at SemEval-2019 Task 4: Hyperpartisan News Detection using ELMo Sentence Representation Convolutional Network. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Youngjun Joo and Inchon Hwang. 2019. Steve Martin at SemEval-2019 Task 4: Ensemble Learning Model for Detecting Hyperpartisan News. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Johannes Kiesel, Florian Kneist, Milad Alshomary, Benno Stein, Matthias Hagen, and Martin Potthast. 2018. Reproducible Web Corpora: Interactive Archiving with Automatic Quality Assessment. *Journal of Data and Information Quality (JDIQ)*, 10(4):17:1–17:25.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jürgen Knauth. 2019. Orwellian-times at SemEval-2019 Task 4: A Stylistic and Content-based Classifier. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Nayeon Lee, Zihan Liu, and Pascale Fung. 2019. Team yeon-zi at SemEval-2019 Task 4: Hyperpartisan News Detection by De-noising Weakly-labeled Data. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Jose G. Moreno, Yoann Pitarch, Karen Pinel-Sauvagnat, and Gilles Hubert. 2019. Rouletabille at SemEval-2019 Task 4: Neural Network Baseline for Identification of Hyperpartisan Publishers. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Osman Mutlu, Ozan Arkan Can, and Erenay Dayanik. 2019. Team Howard Beale at SemEval-2019 Task 4: Hyperpartisan News Detection with BERT. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Duc-Vu Nguyen, Thin Dang, and Ngan Nguyen. 2019. NLP@UIT at SemEval-2019 Task 4: The Paparazzo Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Zhiyuan Ning, Yuanzhen Lin, and Ruichao Zhong. 2019. Team Peter-Parker at SemEval-2019 Task 4: BERT-Based Method in Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Niko Palić, Juraj Vladika, Dominik Cubelić, Ivan Lovrencic, and Jan Snajder. 2019. TakeLab at SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Olga Papadopoulou, Giorgos Kordopatis-Zilos, Markos Zampoglou, Symeon Papadopoulos, and Yiannis Kompatsiaris. 2019. Brenda Starr at SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.

Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. 2018a. The Clickbait Challenge 2017: Towards a Regression Model for Clickbait Strength. *CoRR*, abs/1812.10847.

Martin Potthast, Tim Gollub, Kristof Komlossy, Sebastian Schuster, Matti Wiegmann, Erika Patricia Garces Fernandez, Matthias Hagen, and Benno Stein. 2018b. Crowdsourcing a Large Corpus of Clickbait on Twitter. In *27th International Conference on Computational Linguistics (COLING 2018)*, pages 1498–1507. The COLING 2018 Organizing Committee.

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. 2018c. 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.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait Detection. In

Advances in Information Retrieval. 38th European Conference on IR Research (ECIR 2016), volume 9626 of *Lecture Notes in Computer Science*, pages 810–817, Berlin Heidelberg New York. Springer.

Abdelrhman Saleh, Ramy Baly, Alberto Barrón-Cedeño, Giovanni Da San Martino, Mitra Mohtarami, Preslav Nakov, and James Glass. 2019. Team QCRI-MIT at SemEval-2019 Task 4: Propaganda Analysis Meets Hyperpartisan News Detection. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Saptarshi Sengupta and Ted Pedersen. 2019. Duluth at SemEval-2019 Task 4: The Pioquinto Manterola Hyperpartisan News Detector. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Daniel Shaprin, Giovanni Da San Martino, Alberto Barrón-Cedeño, and Preslav Nakov. 2019. Team Jack Ryder at SemEval-2019 Task 4: Using BERT Representations for Detecting Hyperpartisan News. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Vertika Srivastava, Ankita Gupta, Divya Prakash, Sudeep Sahoo, Rohit R. R., and Yeon Hyang Kim. 2019. Vernon-fenwick at SemEval-2019 Task 4: Hyperpartisan News Detection using Lexical and Semantic Features. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Bozhidar Stevanoski and Sonja Gievska. 2019. Team Ned Leeds at SemEval-2019 Task 4: Exploring Language Indicators of Hyperpartisan Reporting. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Emmanuel Vincent and Maria Mestre. 2018. Crowdsourced Measure of News Articles Bias: Assessing Contributors' Reliability. In *Proceedings of the 1st Workshop on Subjectivity, Ambiguity and Disagreement (SAD) in Crowdsourcing*, pages 1–10.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Chia-Lun Yeh, Babak Loni, and Anne Schuth. 2019. Tom Jumbo-Grumbo at SemEval-2019 Task 4: Hyperpartisan News Detection with GloVe vectors and SVM. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Albin Zehe, Lena Hettinger, Stefan Ernst, Christian Hauptmann, and Andreas Hotho. 2019. Team Xenophilius Lovegood at SemEval-2019 Task 4: Hyperpartisanship Classification using Convolutional Neural Networks. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Chiyu Zhang, Arun Rajendran, and Muhammad Abdul-Mageed. 2019. UBC-NLP at SemEval-2019 Task 4: Hyperpartisan News Detection With Attention-Based Bi-LSTMs. In *Proceedings of The 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.