Doris Martin at SemEval-2019 Task 4: Hyperpartisan News Detection with Generic Semi-supervised Features

Rodrigo Agerri
IXA NLP Group, University of the Basque Country UPV/EHU
rodrigo.agerri@ehu.eus

Abstract

In this paper we describe our participation to the Hyperpartisan News Detection shared task at SemEval 2019. Motivated by the late arrival of Doris Martin, we test a previously developed document classification system which consists of a combination of clustering features implemented on top of some simple shallow local features. We show how leveraging distributional features obtained from large in-domain unlabeled data helps to easily and quickly develop a reasonably good performing system for detecting hyperpartisan news. The system and models generated for this task are publicly available.

1 Introduction

The definition of hyperpartisan according to the Hyperpartisan News Detection shared task at SemEval 2019 (Kiesel et al., 2019) is the following: "Given a news article text, decide whether it follows a hyperpartisan argumentation, i.e., whether it exhibits blind, prejudiced, or unreasoning allegiance to one party, faction, cause, or person".[1] Putting it simply, the task is, given a news article, to decide whether such document is hyperpartisan (true) or not (false). This task is related to the Stance Detection (Mohammad et al., 2016) and automatic detection of fake news (Pérez-Rosas et al., 2018) tasks, which are getting increasing attention within the Natural Language Processing community (Potthast et al., 2018). In this sense, it could be the case that hyperpartisanism is conveyed by some elements of fake news within the article, usually with the objective of spreading propaganda and manipulate readers towards a particular stance on a specific topic.

The SemEval 2019 task 4 aims to address the problem of hyperpartisan news detection at document level, without trying to distinguish specific elements or indicators of hyperpartisanism in each article. Two sets of data were released to participants. The first part (*bypublisher*) is annotated at publisher level. This means that if a publisher is thought to be spreading hyperpartisan news, then all its articles are annotated as hyperpartisan. The *bypublisher* set contains 750K articles divided in 600K documents for training and a validation set of 150K documents. The second part (*byarticle*) has been annotated at article level via crowdsourcing and consists of 645 articles for training and 628 documents for the test. The test set is hidden in TIRA (Potthast et al., 2019) and it is used for the official evaluation scores of the task. It should be noted that, unlike the *byarticle* test set, the *byarticle* training set was not balanced (407 false vs 238 true).

We address this task using an existing document classification system, mostly due to the fact that we joined the task just a week before the final submission deadline. However, and despite the lack of time to implement specific features for the task, we obtained quite good results with a simple and very general feature set in which the most meaningful feature was the use of pre-trained clusters obtained from the English Wikipedia and the Gigaword 5th edition. Out of 42 participants, our official submission obtained 0.737 accuracy whereas the winner of the task scored 0.822.

In addition to our official participation, in this paper we also describe a second round of experiments performed after the official submission deadline. The objective was to establish whether using clusters trained on domain-specific data would improve the results with respect to those obtained by using clusters based on general domain text such as Wikipedia and Gigaword. As it turned out, this second round of experiments allowed us to considerably improve the results

[1] https://pan.webis.de/semeval19/semeval19-web/index.html

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 944–948
Minneapolis, Minnesota, USA, June 6–7, 2019.

(0.761) with respect to our official scores in the task (0.737), confirming that training clusters on domain-specific data, although smaller, helps to address the hyperpartisan news detection task.

2 Methodology

We parsed the given data in XML format extracting the title and the document body for training. We experimented with the original corpus version and with a cleaned (HTML tags removed) and tokenized version. All the pre-processing was done using the IXA pipes tools (Agerri et al., 2014).

Our system learns language independent models which consist of a set of local, shallow features complemented with semantic distributional features based on clusters obtained from a variety of out-of-domain and domain-specific data sources. We show that our approach, despite the lack of hand-engineered, language- and task-specific features, obtains competitive results in the hyperpartisan news detection task.

For the official results we trained only on the *byarticle* training set. The best settings of our system were chosen via 5-fold cross validation. The chosen models and software were uploaded to TIRA (Potthast et al., 2019) to annotate and evaluate the test data. For the official runs, we used pretrained clusters from the Wikipedia and the English Gigaword, as described by Agerri and Rigau (2016).

For the second round of experiments, we used the large *bypublisher* data set and a Fake News Kaggle set[2] in order to train clusters. The motivation was to test whether using data sources closer to the task domain, as opposed to using general text data from Wikipedia and Gigaword, helped to obtained better word representations for this task.

3 ixa-pipe-doc

Our document classification system is *ixa-pipe-doc*, which aims to establish a simple and shallow feature set, avoiding any linguistic motivated features, with the objective of removing any reliance on costly extra gold annotations (POS tags, lemmas, semantics) and/or cascading errors if automatic language processors are used. The underlying motivation is to obtain robust models to facilitate the development of document classification systems for several languages, datasets and domains while obtaining state of the art results.

The system consists of: (i) Local, shallow features based mostly on orthographic, word shape and n-gram features plus their context; (ii) three types of simple clustering features, based on unigram matching; (iii) publicly available gazetteers, such as sentiment lexicons. Specifically, *ixa-pipe-doc* implements, on top of the local features, a combination of word representation features: (i) Brown (1992) clusters, taking the 4th, 8th, 12th and 20th node in the path; (ii) Clark (2003) clusters and, (iii) Word2vec (Mikolov et al., 2013) clusters, based on K-means applied over the extracted word vectors using the skip-gram algorithm. The implementation of the clustering features looks for the cluster class of the incoming token in one or more of the clustering lexicons induced following the three methods listed above. If found, then we add the class as feature. The Brown clusters only apply to the token related features, which are duplicated.

ixa-pipe-doc, as a component of IXA pipes, includes a simple method to combine various types of clustering features induced over different data sources or corpora. This method has already obtained state of the art results in several tasks such as newswire Named Entity Recognition (Agerri and Rigau, 2016) and Opinion Target Extraction (Agerri and Rigau, 2019), both in out-of-domain and in-domain evaluations.

Clusters of words provide denser document representations. Although still a one-hot vector representation, the dimensions of the representation gets reduced to the number of clustering classes used. This is done by mapping the words in the document to the words in each of the clustering lexicons thereby obtaining a denser representations than the traditional one-hot representation based bag of words (Turian et al., 2010).

Finally, *ixa-pipe-doc* learns supervised models via the Maxent algorithm (Ratnaparkhi, 1999). To avoid duplication of efforts, the system uses the Apache OpenNLP project implementation of Maxent [3] customized with the features described in this section.

4 Experiments

We train ixa-pipe-doc with the default parameters, performing 100 iterations with a 5 count cutoff.[4]

[2] https://www.kaggle.com/c/fake-news

[3] http://opennlp.apache.org/

[4] Only features that occur more than 5 times are considered (Ratnaparkhi, 1999).

Features	F1 True	F1 False	Accuracy
token	0.655	0.810	0.755
char26	0.643	0.806	0.749
pref04	0.662	0.810	0.757
token + pref04	0.669	0.809	0.758
token + char26	0.652	0.807	0.752
pref04 + char	0.655	0.812	0.757
(local) Token + char26 + pref04	0.665	0.813	0.759
local + CW600	0.672	0.814	0.763
local + W2VG200	0.674	0.817	**0.766**
local + CW600+W2VG200	0.671	0.816	**0.764**

Table 1: 5-fold cross validation for official results on the *byarticle* training set. CW600: Clark Wikipedia 600 clusters; W2VG200: Word2vec Gigaword 200 clusters.

Features	Accuracy	P	R	F1
Local + W2VG200	**0.737**	0.754	**0.704**	**0.728**
Local + CW600+W2VG200	0.714	**0.773**	0.608	0.680

Table 2: Official results on TIRA test set. CW600: Clark Wikipedia 600 clusters; W2VG200: Word2vec Gigaword 200 clusters.

We only tested three types of **local features** which were already implemented in the system: the current token, the character ngrams of each token (2:6 range) and word prefixes (0-4 characters of each token).

Due to our late arrival to the task, we combined the best local features with our pre-trained clusters from Wikipedia and Gigaword for the official results described in section 4.1. For the second round of experiments of section 4.2, we used the clusters trained using the *bypublisher* and Fake News datasets. The number of clusters trained with each algorithm and data source was the following: 100-800 clusters using the Clark and Word2vec methods, and 1000 classes with the Brown algorithm. The best combination of features were obtained by performing every possible permutation between them in a 5-fold cross validation setting using the *byarticle* training data.

4.1 Official Results

Table 1 provides the 5-fold cross validation results used to choose the two best runs that we submitted for testing on TIRA. As it can be seen, the performance for the *true* and the *false* classes greatly differ. This could be due to the unbalanced nature of the *byarticle* training set or because classifying articles that are hyperpartisan is actually more difficult.

The official results obtained by our system are shown in Table 2. These results show that the main weakness of the system is its lower recall. The local features used usually obtain high precision and lower recall whereas the clustering features reduce sparsity thereby improving the recall. The exception was the Brown clusters, which were detrimental to performance. This is consistent with previous experiments using clusters trained in out-of-domain data (Agerri and Rigau, 2019). Finally, although TIRA did not show the results per class (true or false) we believe that our system reproduced, for the official test data, the behaviour observed in the cross validation experiments.

Therefore, our results seem to indicate that the data used in our pre-trained clusters, Wikipedia and Gigaword, does not allow us to create good word representations for the hyperpartisan news data. Still, it can be said that our official results were promising, obtaining 0.737 versus the 0.822 accuracy of the best system.[5]

4.2 Second Round

This second round of experiments consisted of replacing the out-of-domain cluster lexicons from Wikipedia and Gigaword with those trained on the *bypublisher* and Fake News data. These are the "local + clusters" models in Table 3, which shows the results of performing 5-fold cross validation

[5] https://pan.webis.de/semeval19/semeval19-web/leaderboard.html

Features	F1 True	F1 False	Accuracy
local + W2VHP300	0.675	0.819	0.769
local + W2VFN400	0.670	0.813	0.761
local + W2VHP300+W2VFN400 (clusters)	0.677	0.825	0.773
local + clusters + polarity	0.675	0.824	0.772
local-token + clusters	0.677	0.826	0.774
local-token + clusters + polarity	**0.678**	**0.827**	**0.775**

Table 3: 5-fold cross validation for the second round of results on the *byarticle* training set. W2VHP300: Word2vec Hyperpartisan bypublisher 300 clusters; W2VFN400: Word2vec Fake News 400 clusters.

Features	Accuracy	P	R	F1
local	0.707	**0.768**	0.592	0.669
local + W2VHP300+W2VFN400 (clusters)	0.754	0.719	0.834	0.772
local + clusters + polarity	0.754	0.717	**0.840**	**0.774**
local-token + clusters	0.756	0.731	0.808	0.768
local-token + clusters + polarity	**0.761**	0.734	0.818	0.774

Table 4: Second round results. W2VHP300: Word2vec Hyperpartisan bypublisher 300 clusters; W2VFN400: Word2vec Fake News 400 clusters.

on the *byarticle* training set in order to choose the best models for testing.

Furthermore, Table 3 reports the results of three additional experiments: (1) adding three polarity lexicons to the *local + clusters* model; (2) removing the current token feature from the *local* feature set (*local-token + clusters*) and, (3) adding the three polarity lexicons to experiment (2). The motivation of removing the current token feature was to see if that helped the system to generalize better over unseen words. The features based on polarity add a polarity value (positive or negative) if a word in training or testing gets matched in one of the three polarity lexicons used. More specifically, we used three different lexicons (Hu and Liu, 2004; Riloff and Wiebe, 2003; Mohammad et al., 2009), resulting in three different features for each token. As in the previous section, in this phase we realized that our system consistently performs much better, for every experiment, for the "false" class. Experimenting with a balanced training set is left for future work.

As we expected, using domain-specific clustering-based word representations substantially improved the recall results, which in turn led to substantial improvements in terms of accuracy and F1 score. This improvements are reflected also on the evaluation on the test data hidden in TIRA. Thus, Table 4 reports considerable gains obtained by using clustering features in terms of recall with respect to the model based on local features only. The final reported score is 0.761 in accuracy, still lower than the top score in the task (0.822), but a significant result obtained by the simple method of providing better word representations (closer to the task domain) based on clustering. The improvements of this second round of experiments are larger in terms of F1 score, which goes up to 0.774, closer to the winner's F1 score of 0.809.

Most importantly, our experiments show that our system, even though generic, simple and lacking task-specific features, allows to easily obtain competitive results for a document classification task such as hyperpartisan news detection.

5 Concluding Remarks

This paper describes our first experiments on the Hyperpartisan News Detection task organized at SemEval 2019 (Kiesel et al., 2019). We aim to improve our work in the task by using other techniques such as denser word representations based on continuous vectors (word embeddings) and deep learning architectures for document classification. We would also like to investigate the relation with other tasks such as Stance Detection (Mohammad et al., 2016) and automatic detection of fake news (Pérez-Rosas et al., 2018). The system and models can be found in `https://github.com/ixa-ehu/ixa-pipe-doc`.

Acknowledgments

This work has been supported by Spanish Ministry of Economy and Competitiveness (MINECO/FEDER, UE), under the project CROSSTEXT (TIN2015-72646-EXP) and the Ramon y Cajal Fellowship RYC-2017-23647.

References

Rodrigo Agerri, Josu Bermudez, and German Rigau. 2014. IXA pipeline: Efficient and ready to use multilingual NLP tools. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2014)*.

Rodrigo Agerri and German Rigau. 2016. Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

Rodrigo Agerri and German Rigau. 2019. Language independent sequence labelling for opinion target extraction. *Artificial Intelligence*, 268:85–95.

Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Alexander Clark. 2003. Combining distributional and morphological information for part of speech induction. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 59–66. Association for Computational Linguistics.

M. Hu and B. Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

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.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

S. Mohammad, C. Dunne, and B. Dorr. 2009. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 599–608.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41. Association for Computational Linguistics.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2018. Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401. Association for Computational Linguistics.

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.

Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine learning*, 34(1-3):151–175.

E. Riloff and J. Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the International Conference on Empirical Methods in Natural Language Processing (EMNLP'03)*.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394, Uppsala, Sweden. Association for Computational Linguistics.