Team yeon-zi at SemEval-2019 Task 4: Hyperpartisan News Detection by De-noising Weakly-labeled Data

Nayeon Lee[*], Zihan Liu[*], Pascale Fung
Center for Artificial Intelligence Research (CAiRE)
Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
`[nyleeaa, zliucr].connect.ust.hk, pascale@ece.ust.hk`

Abstract

This paper describes our system submitted to SemEval-2019 Task 4: Hyperpartisan News Detection. We focus on removing the inherent noise in the hyperpartisanship dataset from both data-level and model-level by leveraging semi-supervised pseudo-labels and the state-of-the-art BERT model. Our model achieves 75.8% accuracy in the final by-article dataset without ensemble learning.

1 Introduction

With the ever-growing usage of internet, the problem of fake news that spreads in a destructive speed has attracted many attention. Fake news is a kind of news that is typically inflammatory, extremely one-sided (hyper-partisan) or untruthful to mislead the public into having distorted belief.

Previous works attempted to solve fake news problem from various aspects, ranging from knowledge-based (Wu et al., 2014; Shi and Weninger, 2016; Lee et al., 2018) to style-based (Wang, 2017; Potthast et al., 2018). There are some publicly available fake news datasets, however, often too small in size to be suitable for neural approaches (Horne and Adali, 2017; Pérez-Rosas et al., 2017). Recently, the organizers of SemEval2019 Task 4 (Kiesel et al., 2019) have released large-scale dataset to address fake news detection as a hyper-partisan news detection problem. The task is to determine whether a given article is hyper-partisan (extremely right-wing or left-wing) or not (mainstream). Such task will allow for pre-screening of semi-automatic fake news detection, and more importantly, bring us one step closer to solving fully automated fake news detection.

Initially, we focused on learning and utilizing useful features such as topic and sentiment information. Considering the purpose of hyper-partisan news, we believed that the stance on politically sensitive topics would be crucial in determining hyperpartisanship. However, experiments showed that the dataset contains some inherent noise that acted as a big barrier to learning a good classifier: 1) *noisy text* inputs from an article that contain domain-specific (i.e. political) words, slangs and spelling mistakes which are likely to be out of vocabulary (OOV). 2) *noisy labels* that mainly resulted from using publisher-level information for labeling articles (i.e. all articles from left/right-wing publishers are labeled as "hyper-partisan". For more detail, refer to Section 2).

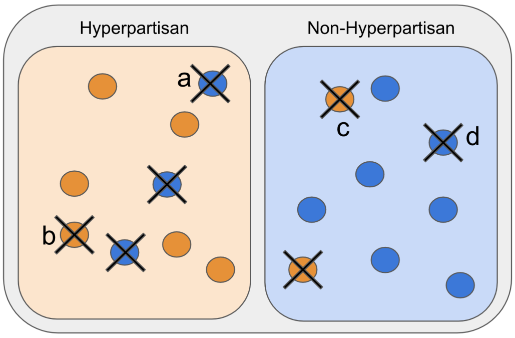

Figure 1: Illustration of filtering. Sample *a* and *c* are removed as pseudo-label ≠ by-publisher label. Sample *b* and *d* are removed as their prediction confidence was below threshold.

Nevertheless, human-labeled large-scale dataset creation is a very expensive and time-consuming task, thus, it is crucial to find a better way to utilize this weakly-labeled dataset. Therefore, we experimented with reducing noise to help models learn better. In our work, we apply a semi-supervised pseudo-labeling to de-noise the dataset (Figure 1) and leverage the state-of-the-art pre-trained BERT (Devlin et al., 2018) to obtain a better representation of the noisy input.

[*] These two authors contributed equally.

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 1052–1056
Minneapolis, Minnesota, USA, June 6–7, 2019. ©2019 Association for Computational Linguistics

2 Data Analysis

Label Items	Train Set %	Val Set %
right	25%	25%
right-center	7.1%	8.8%
left	25%	25%
left-center	11.7%	15.7%
least bias	**31.2%**	**25.5%**
all	100%	100%
hyperpartisan	50%	50%
mainstream	50%	50%

Table 1: Data statistic of hyperpartisan and political orientation on by-publisher dataset.

We use a publicly available dataset "SemEval 2019 Task 4 - Hyperpartisan News Detection" [1] that are labeled in two different ways - publisher level and article level.

- Publisher-level (by-publisher): A total of 750K articles are labeled based on the political orientation of the publisher, without considering the content. It has an equal ratio (375K/375K) between hyperpartisan and non-hyperpartisan. Among the hyperpartisan samples, there's an equal ratio (187.5K/187.5K) between right and left political orientation.

- Article-level (by-article): A total of 645 articles labeled on article-level by checking the actual content. The data contains only articles for which a consensus among the crowdsourcing workers existed. Of these, 238 (37%) are hyperpartisan and 407 (63%) are not.

2.1 Discussion on the Inherent Noise

By using human judgment, we discovered that some article samples did not always have the correct labels. Since the political orientation of the publisher was used as a sole criterion for the labels, such labeling noise is not surprising. It cannot be guaranteed that all articles from a hyperpartisan publisher are hyper-partisan. Another possible reason for such noise could be from not having enough non-hyper-partisan publishers (i.e. The percentage of "least bias" label items in Table 1 is not 50%), thus, treating news from "right-center" and "left-center" publishers also as non-hyper-partisan.

3 Methodology

In this section, we describe how we did de-noising in our system in Figure 2. Our system consists of two steps: 1) Obtaining de-noised by-publisher dataset by leveraging clean by-article dataset. 2) Leveraging the de-noised by-publisher dataset and pre-trained BERT to train our final model. Note that our code is publicly available for reproducibility [2].

3.1 Step 1: Filter Noise by Leveraging Pseudo-labeling

To deal with the noise in the labels, we utilize pseudo-label for filtering out noisy labels from data-level (Figure 1). Pseudo-labeling is one of the semi-supervised learning methods, which approximates the labels of unlabeled data by using a model (M) trained on the labeled dataset. Originally, pseudo-labeling directly takes the prediction from the model M as the label. This could result in the final model trained on both human-labeled and pseudo-labeled to be bounded by the accuracy of the model M.

To avoid this problem: 1) We use the original by-publisher label as the constraint. We filter out data points that have a mismatch in the by-publisher label and pseudo-label to obtain cleaner by-publisher. 2) To be robust to the errors made by the model M, we set some thresholds to only use pseudo-labels with relatively high confidence. We only consider prediction scores that is bigger/smaller by $margin = 0.2$ than the mid-value (0.5). By doing so, we can filter out noisy labels with the guarantee that the noise level would be at worst kept the same; the size of our de-noised dataset is approximately 32K for both labels, which is 8.5% of original data. Note that in our system, the model M is a binary classifier trained on top of fine-tuned BERT (refer to step 2) using clean by-article dataset.

3.2 Step 2: Obtain Better Input Representation using BERT

The article texts are noisy with a lot of political words, slangs, and even spelling mistakes, many of which are out of vocabulary (OOV) and harmful to the sentence-level and article-level representation learning. We leverage state-of-the-art pre-trained language representation model BERT to

[1] https://zenodo.org/record/1489920#.XAAoMJMzYWq

[2] https://github.com/zliucr/hyperpartisan-news-detection

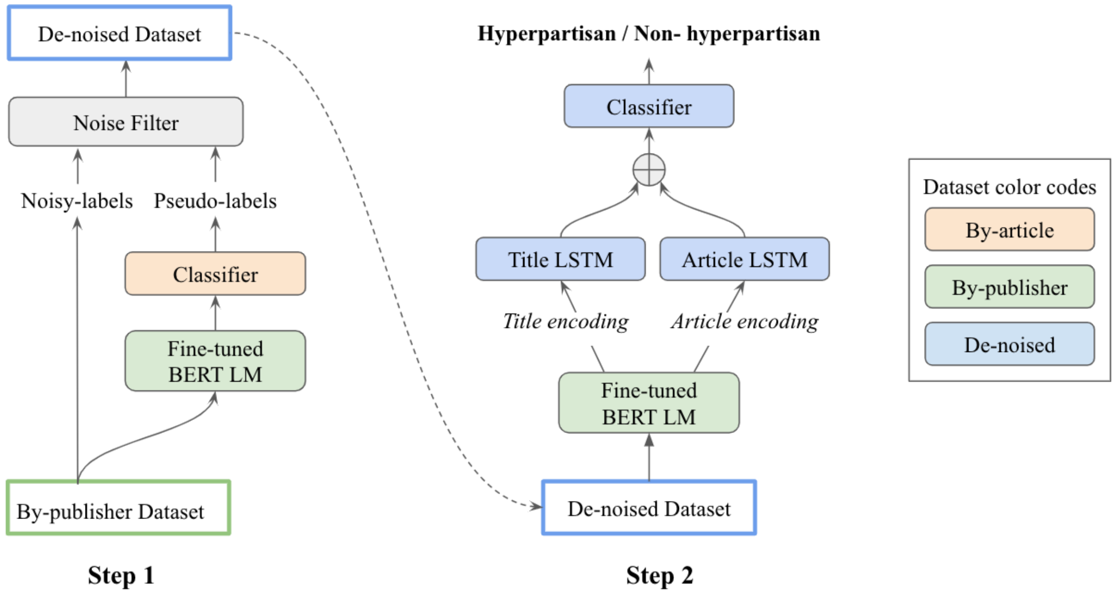

Figure 2: Architecture of the proposed system. Colors represent the dataset used to train the corresponding model.

eliminate OOV problem, since it uses byte-pairs vocabulary, and for a better input representation.

Since the pre-trained BERT model is trained on BooksCorpus and Wikipedia which are not directly relevant to news, we fine-tune the BERT, as in original paper, using our by-publisher news dataset to learn a better representation for our data domain. We build our proposed model by adding title LSTM and article LSTM on top of the fine-tuned BERT model to extract features that are concatenated and fed into the final binary classifier. We train our final classifier using the filtered dataset from Step 1.

4 Experiments and Analysis

4.1 Experimental Setup

We use BERT$_{BASE}$ model from (Devlin et al., 2018) which has 12 layers (i.e., Transformer blocks) with a hidden size of 768 and 12 self-attention heads. In step 1, the parameters of BERT model were fixed after fine-tuned on by-publisher datset, then we trained classifier on by-article dataset by using 16 batch size. We used 10-fold cross-validation to choose the parameters of the classifier, since the size of by-article dataset is small. In step 2, we used 16 batch size to train our LSTM for article model with a hidden size of 300 and LSTM for title model with a hidden size of 100. The classifiers in step 1 and 2 both consist of two linear layers with ReLU and batch normalization in between.

For the evaluation metric, we mainly considered accuracy and F1 score as the main indicator of performance. For analysis purpose, we also report precision and recall. In the competition, there were two types of test sets (i.e. by-publisher test set and by-article test set). However, all of the reported results are obtained from the by-article test set for fair and correct comparison.

4.2 Results

We ran the experiment on 3 baseline models for comparison and simple ablation study of our approach, and the results are presented in Table 2.

- **2 LSTM + Attention + Fine-tuned Classifier** ($LSTM_{ft}$)
  A baseline model consisting of 2 LSTM models (one for the title, and another for the article) with attention layers and a multi-layer perceptron (MLP) as a classifier on the top. It was trained on by-publisher dataset directly, then fine-tuned using the by-article dataset.

- **Pre-trained BERT+Classifier** ($BERT_{pt}$)
  This model uses the original pre-trained BERT model to encode both article and title, which get fed into multilayer perceptron (MLP) to predict the hyper-partisanship of the given article. The parameters of the BERT model was fixed when training the MLP classifier on the by-article data.

Models	Accuracy	Precision	Recall	F1
$LSTM_{ft}$	0.6258	0.5838	**0.8758**	0.7006
$BERT_{pt}$	0.5669	**0.8621**	0.1592	0.2688
$BERT_{ft}$	0.6592	0.8378	0.3949	0.5368
$BERT_{ft}$ + De-noise	**0.758**	0.744	0.7866	**0.7647**

Table 2: Results of our model and other baseline models on the final by-article test set.

- **Fine-tuned BERT+Classifier** ($BERT_{ft}$)
  For this model, everything is kept the same as $BERT_{pt}$ except for the fact that pre-trained BERT was fine-tuned using by-publisher dataset.

Firstly, we can observe that simply using pre-trained BERT ($BERT_{pt}$) to represent input cannot out perform LSTM model entirely trained on hyperpartisan dataset. However, by fine-tuning BERT using our dataset ($BERT_{ft}$), we gain improvement in performance by approximately 10% in accuracy, outperforming $LSTM_{ft}$ by $\approx$ 3%. Hence, we can infer that by injecting some domain-specific data into the original BERT, we can obtain an improved text representation for solving our task. Note that the model sizes for Pre-trained BERT + Classifier and Fine-tuned BERT + Classifier are the same.

Secondly, by training the same fine-tuned BERT model on the de-noised dataset mentioned in Section 3.1, we observed a big improvement in accuracy, F1 and recall by $\approx 10\%$, $\approx 23\%$ and $\approx 40\%$ respectively. This clearly illustrates the power of de-noising the dataset using pseudo-labels as auxiliary reference label. We also would like to emphasize that we did not use any ensemble learning or tricks, which normally gives extra $1 - 2\%$ gain in the final performance. Our system ranked 11 out of 43 teams that participated.

Lastly, we would mention that our $LSTM_{ft}$ model is a strong baseline because it was able to achieve a high score in the by-publisher test set by obtaining 0.663 and 0.694 for accuracy and F1 respectively (rank 5/28).

4.3 Interesting Analysis

Although our current system does not make direct use of topic information, we present an interesting result obtained while experimenting with topic modeling for hyper-partisanship detection. We used Latent Dirichlet allocation (LDA) for topic modeling, and the results empirically showed interesting relationships between topics and hyper-partisanship. Sensitive topics such as war and political parties tend to have more hyperpartisan news than neutral-topics such as school and sports games. We believe that leveraging such information would be helpful in future works.

5 Related Works

In this part, we briefly review the prior work in language representation as well as the semi-supervised learning method we used.

5.1 Language Representation

(Kiros et al., 2015) tried to learn sentence embedding by reconstructing the surrounding sentences of an encoded passage. (Peters et al., 2018) proposed to extract context-sensitive features from a language model. (Devlin et al., 2018) jointly conditioned on both left and right context and obtained state-of-the-art results on eleven natural language processing tasks.

5.2 Semi-supervised Learning

(Triguero et al., 2015) provided a survey of self-labeled methods for semi-supervised classification. (Zhu and Goldberg, 2009) showed self-labeled techniques are typically divided into self-training and co-training. (Lin et al., 2018) proposed semi-supervised learning to leverage a small amount of user-comment data to train a model and then expand the dataset by that trained model.

6 Conclusion

To conclude, we successfully removed noise from data-level and model-level by utilizing pseudo-labels and state-of-the-art BERT. Compared to other baselines, our de-noised model managed to outperform all, and achieve rank 11 from 42 teams. Since the cost of manual labeling fake news data is expensive, our approach to obtain cleaner and larger dataset by leveraging smaller but clean dataset is meaningful.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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*.

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.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Nayeon Lee, Chien-Sheng Wu, and Pascale Fung. 2018. Improving large-scale fact-checking using decomposable attention models and lexical tagging. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1133–1138.

Zhaojiang Lin, Genta Indra Winata, and Pascale Fung. 2018. Learning comment generation by leveraging user-generated data. *arXiv preprint arXiv:1810.12264*.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2017. Automatic detection of fake news. *arXiv preprint arXiv:1708.07104*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

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.

Baoxu Shi and Tim Weninger. 2016. Fact checking in heterogeneous information networks. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 101–102. International World Wide Web Conferences Steering Committee.

Isaac Triguero, Salvador García, and Francisco Herrera. 2015. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information systems*, 42(2):245–284.

William Yang Wang. 2017. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*.

You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2014. Toward computational fact-checking. *Proceedings of the VLDB Endowment*, 7(7):589–600.

Xiaojin Zhu and Andrew B Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.