[Draft] Fairly Fake: Learning fair classifiers for fake news detection

Zeerak Waseem
University of Sheffield
United Kingdom
z.w.butt@sheffield.ac.uk

Kalina Bontcheva
University of Sheffield
United Kingdom
kbontcheva@sheffield.ac.uk

Andreas Vlachos
University of Cambridge
United Kingdom
andreas.vlachos@cst.cam.ac.uk

Abstract

Fake news detection has received an increasing amount of attention from the Natural Language Processing community over the past few years, where several papers find that the use of meta-data for documents can aid text-based models in detecting fake news. In this paper, we focus on linear classifiers for fake news detection and show that they encode bias present in the training set. We address this by extending recent work on fairness in machine learning and apply it to text classification for the fake news detection task. The method encourages fairness by incorporating a set of linear constraints into the learning objective that seek to limit the model's reliance on features corr with protected attributes, such as political affiliation of the speaker. We demonstrate that by using this method, we can reduce political bias encoded in the model, resulting in a reduction of the difference in error rates between each class by 57.45% while increasing the error rates for all classes.

1 Introduction

Over recent years fake news detection has received an increasing amount research attention. In the Natural Language Processing (NLP) community, a number of papers have attempted at identifying misinformation and fake news using a combination of text (Thorne and Vlachos, 2018; Pérez-Rosas et al., 2018) and meta-data (Wang, 2017; Karimi et al., 2018). However as language production may encode demographic and stereotypes, models produced using textual data may encode the same biases (Bolukbasi et al., 2016; Gurses et al., 2018). For fake news detection, in using metadata (Wang, 2017; Karimi et al., 2018; Long et al., 2017) for detection may further bias trained models to learn correlations between demographic attributes and the output label. In this paper, we extend a rebiasing method proposed by Agarwal et al. (2018) to text classification. The method operates on machine learning models and seeks to reduce the impact of confounding variables through constrained optimization of the model's weights. We examine the political bias learned by three different linear classifiers on the fake news detection dataset proposed by Wang (2017), finding that models that do not undergo re-biasing rely on features which strongly correlate with political affiliation, and that we can re-bias our models for fake news detection along the axis of politiical affilation. We find that the impact of re-biasing our models results in a loss of accuracy from 63.13% to 52.25% while reducing the difference in error rates between the classes by 57.45%.

2 Classifiers with Fairness Constraints

We address the challenge of rebiasing a classifier by applying the method proposed by Agarwal et al. (2018). It seeks to rebias a classifier as the result of a constrained optimisation, in which a saddlepoint which adheres to the constraints is computed. Specifically, for each protected attribute we iteratively compute a lagrangian multiplier, which we use to constrain the model. Formally, we can express our constraints as

$$\min_{Q \in \Delta} \widehat{err}(Q) \text{ subject to } \boldsymbol{M}\boldsymbol{\mu}(Q) \leq \mathbf{c}$$

where $\mathbf{c}$ and $\mathbf{M}$ express the linear constraints as a matrix and vector, Q is a randomized classifier, and $\boldsymbol{\mu}$ is a conditional moment. Agarwal et al. (2018) treat the saddlepoint computation as a two-player game, in which a λ-player adjusts the weights of a classifier while the Q-player computes the constraint violations. The algorithm iteratively updates the lagrangian multipliers until it reaches a point where "neither player can obtain more than ν by changing their choice" (Agarwal et al., 2018). By iteratively producing classifiers

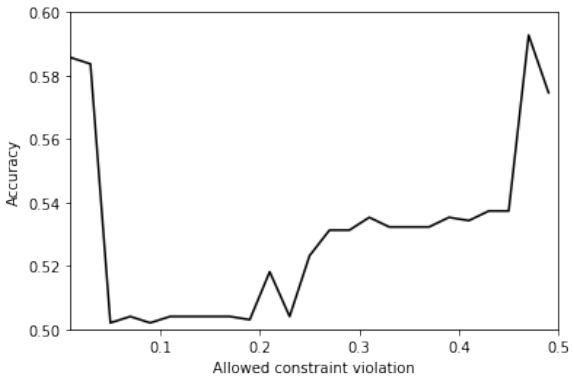

Figure 1: Development of accuracy score given constraint strength ϵ

with different manipulations of the weights as a function of the constraint violations, we obtain a fairer classifier as each iteration of the weights is a step towards removing the influence of the of the protected class.

In order to find a balance between the strength of the fairness constraints and the performance of the model, we perform a grid search over the allowed constraint violation ϵ $(0.01 - 0.5)$, the convergence threshold ν $(0.01 - 0.5)$, and the learning rate η $(10^{-(5-20)})$. We find that only the strength of the allowed constraint violation influences the model (see Figure 1).

For all experiments we set $\epsilon = 0.263$.

3 Evaluation and Datasets

3.1 Evaluation Metrics

While we retain a focus on overall accuracy for our models as a guide, we also consider the error rate computed for class 0 (replacing $y_i = 0$ with $y_i = 1$ for class 1)

$$\frac{1}{|Y|} \sum_{i \in X} \mathbf{1}\{y_i \neq \hat{y}_i \wedge y_i = 0\}$$

where C is the set of classes and X is the set of documents that are being predicted on, and Y is the set of labels.

We look at the development of feature weights of the 10 most predictive features for each class for a model predicting the protected attributes. Finally, we compute the LIWC categories of the 1000 most heavily weighted features for our biased and fair classifiers.

3.2 Dataset

We sample our datasets from the dataset published by Wang (2017) (see **??** for class distribution). The original dataset consists of $12,800$ manually annotated short statement by Politifact.[1] Aside from the labels of factual correctness, the dataset also provides additional information about each document, including the name of the speaker or organisation and the party affiliation of the speaker. The dataset however, does not provide justification or support for the labels - as such the dataset can only be used for text classification rather than evaluating identifying linguistic, topical, or other relationships which may distinguish truthful information from misinformation. In our experiment predicting misinformation as a binary classification task, so we map the following classes to our positive label: "pants-on-fire", "'barely-true", and "false"; and "true", "mostly-true", and "half true" to our negative class. We hypothesize that a classifier may learn biases due to differences in lexical choices and language production which are expressed along the axis of protected attributes.

3.2.1 Selecting a protected attribute

In any consideration of implementing fair classifiers, it is necessary to pre-select a protected variable which you want to address. While the dataset comes with plenty of attributes which could be considered: gender, state in which the speaker is located or represents, their job, the venue of the statement, the subject on which they're speaking, who the speaker is, which party they belong to, and the history of the evaluation of correctness of their past statements. The history of correctness aside, each of these attributes are potential candidates for fairness as we do not want them to influence whether a statement is predicted to be true or false, e.g. whether a statement is given in a television interview or a radio interview should not influence the prediction. In our consideration of these attributes, two stood out as particularly important: Party affiliation and gender of the speaker (derived from their names).

Gender We sample gender by looking up individual Wikipedia pages for the speakers. We limit the set of potential speakers to those who appear in our sample of documents with a Democratic or Republican party affiliation. We choose to use this subset as the Wikipedia pages will often either directly state gender (e.g. congresswoman)

[1] www.politifact.com is an organisation which fact-checks the statements from politicians, organisations, and rumours that circulate on social media platforms.

Feature Set	Feature	Accuracy	Precision	Recall
n-grams	Unigrams	63,13	0.62	0.63
	Unigrams + Bigrams	**64.04**	0.63	**0.64**
LIWC	LIWC	59.59	0.36	0.60
Sentiment	Sentiment	56.66	0.36	0.60
Dependency Feats	Arc	62.32	0.52	0.51
	Children	62.72	0.61	0.63
	Head	61.31	0.60	0.61
	Arc Count	59.39	0.58	0.59
	Dependency Features	60.70	0.60	0.61
Reading Ease Feats	Dale-Chall	59.49	0.54	0.59
	Sentence Length	59.79	**0.76**	0.60
	Avg. Word Length	59.39	0.46	0.59
	Reading Ease Features	59.09	0.52	0.59
Feature Combinations	Reading Ease Feats + Unigram	63.63	0.63	**0.64**
	Reading Ease Feats + Unigram + Bigram	62.92	0.62	0.63
	Dependency Feats + Unigram	61.01	0.60	0.61
	Dependency Feats + Unigram + Bigram	61.71	0.61	0.62
	Unigrams + LIWC + Sentiment + Dependency Feats + Reading Ease Feats	61.31	0.61	0.61
	LIWC + Unigram	63.33	0.62	0.63
	LIWC + Unigrams + Bigrams	61.51	0.60	0.62
	All Features	62.02	0.61	0.62

Table 1: Feature Exploration using Logistic Regression

or do so indirectly ("He was first elected..."). As not all speakers in the dataset are human[2], thus our dataset is further limited to $9,291$ documents ($1,639$ women and $7,652$ men). One limitation of using a party affiliation sample is that our gender representation operates on the binary scale, as there are no instances of members of the political parties that identify as non-binary and all persons in our dataset are assumed to be cisgender in lack of evidence of other. Due to the majority of speakers in the dataset being male, we expect that there is a heavy bias towards men producing misinformation.

Party Affiliation To investigate whether a fake news classifier learns features that are indicative of party affiliation, we restrict our dataset to only include documents which also contain party affiliation. We further restrict our data sample by only using documents that have an affiliation to the Democratic party or the Republican party ($4,150$ by democrats and $5,687$ by republicans) as the third most frequent party affiliation (independent) is only associated with 149 documents. Thus the resulting dataset comprises 9.8K documents, split into $7,855$ for training, 992 validation, and 990 documents in our test set. We retain the splits proposed by (2017).

[2]Other entity types were: Organisations and cats

3.3 Models

We consider three different machine learning models: Support Vector Machines (SVM) with a linear kernel, Logistic Regression, and Random Forest. For each of the models we use L2 regularization and perform grid-search to determine the regularization used (L1 or L2) and the inverse regularization strength. We set use the same parameters for both the fair and unfair models. We find that all models prefer L2 regularization with an inverse regularization strength of 0.1.

We examine several feature sets including n-grams (1-2), dependency trees, sentiment, readability scores, and LIWC. We find that there is little benefit in of any features beyond n-grams (see Table 1). Additionally, we see that using the entire dataset rather than the subset of documents which have declared party affiliation similarly yields negligible differences in performance in terms of accuracy. Therefore we settle on unigrams due to their relative predictive power and the speed of training models.

Baselines We produce our baseline for the unfair classifiers by replicating the Logistic Regression classifier used in Wang (2017) and retrain it using binary labels. We additionally compute baseline for our fair classifier (blind), by applying the notion of fairness through unawareness (FTU) (Grgic-Hlaca et al., 2016). We compute the blind classifier by training a classifier to predict the protected attribute, we then train a classifier to predict

fake news, removing all instances of the top 200 features for each class from our dataset.

Fair Classifiers For our fairness method, we perform a grid-search over the allowed constraint violation (ϵ) and the convergence threshold for the duality gap (ν). We find that for our dataset, only changing the allowed constraint violation (ϵ) accounts for changes in performance with the highest accuracy. As we wish to balance the strength with which we constrain our model and its performance, we set $\epsilon = 0.26$ for all experiments.

3.4 Results

We evaluate our models along several different axis. We use multiple qualitative evaluations: We consider the development of feature weights in our models for the top 10 features which predict our protected attribute; we compute the LIWC categories invoked by the 200 most discriminative features for each class for the fair and unfair versions of each model. Quantitatively we also consider two views into our models' performances: We examine the rate of misclassification, the precision, and the recall for each class on one hand and the compare accuracy for the models on the other hand.

Gender We omit the results for gender, as we find that the fake news classifier does not learn weights that overlap with gender. Thus, for this dataset there seems to be little evidence of a gender bias.

Party Affiliation Considering the results in Table 2 we notice that by removing access to the 400 most discriminative features for predicting political party there is an increase in classification performance for the task of fake news detection. Comparing this to the performance of the unfair fake news classifier, we can extrapolate that the unfair classifier is indeed influenced by party affiliation.

We notice that the performance of the fair classifiers show a marked drop in predictive power of our classifiers, however, we also notice that the difference in error rates sees more than a 50% drop, suggesting that our fair model is in fact more fairly wrong. Similarly, we while the accuracy for both classes drop, we see a small decrease in the difference in accuracy for each class (see Table 2). Further, we apply our fair and unfair classifiers to predict party affiliation. We map predictions of "republican" to "fake" and predictions of "democrat" to true. Here we notice a large drop in predictive performance, where the unfair classifier performing slightly worse than random chance and the fair classifier performing well below chance, suggesting that the learned feature weights are detrimental to predicting party affiliation.

3.5 Feature Analysis

Beyond quantification of the performances of our models, we also consider our models qualitatively by examining the feature development of our models as they are iterated from an unconstrained model to models which have undergone several iterations of refining constraints. Specifically, we examine the LIWC categories invoked by the top 200 features for each class for our fair and unfair models, and compare these with those invoked by the 200 most predictive features for each class for our classifier predicting the protected attributes. Our other entry into our analysis is examining the development of feature weights as we train our fair classifier.

Feature Tracking To estimate the impact of applying fairness constraints, we identify the features in the intersection of features that are highly predictive of misinformation and highly predictive of party affiliation. We then identify our selected features in our constrained model, and examine the change to these features as a set. E.g. "Obamacare" is a highly predictive for republicans, while "companies" is highly predictive for democrats. Considering the top 200 most predictive features for republicans and democrats respectively, we see that there is a marked change in the weight of our features as (see Table 3 for examples of the development feature weights). For these features, we see a mean standard deviation of 0.1527 and 0.1693 for republican and democrat predictive features, respectively.

Considering the 40 most predictive features for predicting party affiliation for each model type, we see a pattern of minor changes to the feature value for a number of features, with a small number of features that have more extreme weight changes. In Figure 2, we illustrate the development of the 10 most predictive features for party affiliation.[3]

LIWC To gain an intuition into how the changes of feature weights are reflected we examine the

[3]We limit our visualisation to 10 most predictive features for visual clarity.

	Accuracy (overall)	Precision	Recall	F1-score	Accuracy (Class0)	Accuracy (Class1)	Error Rate (Class0)	Error Rate (Class1)
Blind (dem/rep)	0.7745	0.78	0.77	0.77	0.6318	0.88211	0.0672	0.1582
Unfair LR (Fake News - all data)	0.6305	0.63	0.63	0.63	0.4982	0.7317	0.1519	0.2174
Unfair SVM (Fake News - all data)	0.6024	0.60	0.60	0.60	0.5215	0.6643	0.1901	0.2073
Unfair RF (Fake News - all data)	0.6250	0.62	0.63	0.60	0.3669	0.8225	0.1005	0.2743
Random Baseline (Fake News - all data)	0.5003	0.51	0.50	0.50	0.4946	0.5048	0.2805	0.2190
Unfair LR (Fake News - dem/rep)	0.6313	0.62	0.63	0.62	0.4525	0.75254237	0.1474	0.2212
Unfair SVM (Fake News - dem/rep)	0.6020	0.60	0.60	0.60	0.5175	0.6593	0.2030	0.1949
Unfair RF (Fake News - dem/rep)	0.6272	0.61	0.63	0.61	0.3750	0.7983	0.1202	0.2525
Random Baseline (Fake News - dem/rep)	0.5040	0.52	0.50	0.51	0.5	0.5067	0.2939	0.2020
Unfair Pretrained LR (Dem/rep prediction)	0.5202	0.57	0.52	0.51				
Unfair Pretrained SVM (dem/rep prediction)	0.5292	0.56	0.53	0.53				
Unfair Pretrained RF (dem/rep prediction)	0.5060	0.58	0.51	0.48				
Random Baseline (Pretrained Dem/rep prediction)	0.5252	0.54	0.53	0.53				
LR (Dem/rep prediction)	0.6676	0.66	0.67	0.66	0.8086	0.4682	0.2202	0.1121
SVM (Dem/rep prediction)	0.6434	0.64	0.64	0.64	0.6965	0.5682	0.1787	0.1777
RF (Dem/rep prediction)	0.6131	0.60	0.61	0.59	0.8172	0.3243	0.2898	0.1070
Random Baseline (Dem/rep prediction)	0.4717	0.49	0.47	0.48	0.4672	0.4780	0.2161	0.312
Fair Fake News LR (dem/rep)	0.5252	0.52	0.53	0.52	0.36	0.6372	0.2161	0.2585
Fair Fake News SVM	0.4808	0.49	0.48	0.48	0.39	0.5423	0.2727	0.2464
Fair Fake News RF	0.5909	0.58	0.59	0.58	0.415	0.7101	0.1727	0.2363
Random Baseline (Fair LR) - Same as Unfair LR	0.5040	0.52	0.50	0.51	0.5	0.5067	0.2939	0.2020
Fair Pretrained LR (Dem/rep prediction)	0.3575	0.38	0.36	0.35				
Fair Pretrained SVM (Dem/rep prediction)	0.3676	0.39	0.37	0.37				
Fair Pretrained RF (Dem/rep prediction)	0.4959	0.54	0.50	0.49				

Table 2: Results of our models

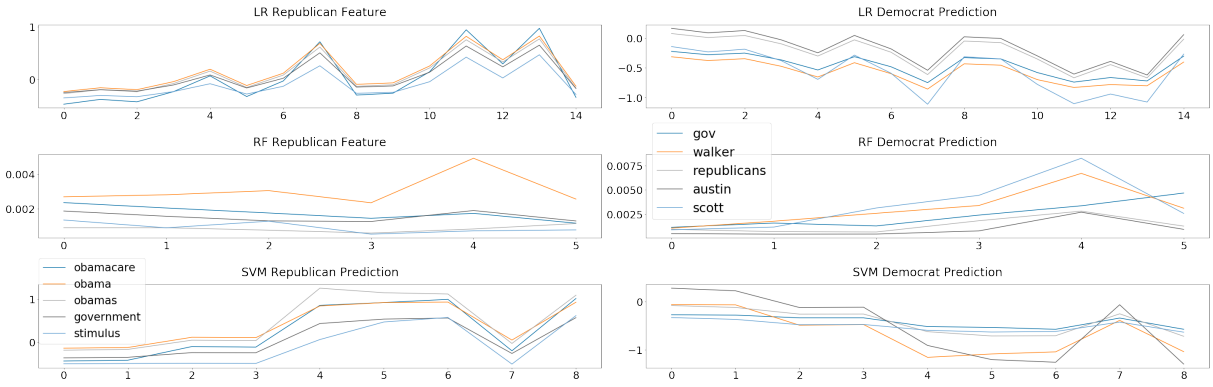

Figure 2: Development of 10 most predictive features for predicting party affiliation.

	Unfair Model (Iter 0)	Iter 1	Iter 2	Iter 3	Iter 4	...	Iter 14
Illegal (republican predictor)	-0.21896608954137983	-0.1583656511853084	-0.18890741319727355	-0.06198869208056063	0.12987212545092477	...	-0.13369198549419622
Undocumented (Democrat predictor)	0.3146494568038171	0.27226071553939585	0.293873402573684	0.2013875747697955	0.054535611380285	...	0.254467618770179

Table 3: Changes in feature weights for a Logistic Regression model in the first 5 (of 15) iterations and the final iteration in each model from unfair to fair.

changes in which categories of the Linguistic Inquiry and Word Count (LIWC) categorisation are activated for the 1000 most highly weighted features for each model. In Table 4 we see the 10 most frequent LIWC categories that are invoked by each classifier.

Considering most frequent LIWC categories activated, we see that there are significant changes in the activate LIWC categories as the activated categories for the "True" class show a higher emphasis on impacts to personal life through the increased attention to "Relativity" "Personal Concerns", and "Affect Words". For the "False" class on the other hand we see that there is a greater attention to the categories "Grammar Other" and "Function Words". In fact, we observe that the activated LIWC features for each class in the unfair classifiers partially reflect the categories for the opposite class in the fair classifier. Thus, the activated LIWC categories sugges that through our process of re-biasing our models, the models learn to rely more heavily on classes of features that the unfair classifiers learned were indicative of the opposite class.

4 Related Work

In our consideration of different approaches towards computing fair machine learning classifiers for natural language processing, we considered different approaches but ultimately landed on applying the reductions method proposed by Agarwal et al. (2018). Amongst others, we considered counter-factual fairness as an approach (Kusner et al., 2017), though the issue of dialectical variety and correlations with labels prevented us from applying it as generating a counter-factual scenario might not be possible without dialectical translation. As the dataset we have chosen, may contain dialectical correlations with political affiliation, as there are traditionally republican-leaning states and traditionally democrat-leaning states, it would be necessary to consider not only the statement but also the dialect. We argue, that for NLP tasks which consider demographic attributes which coincide with difference in language production, generating the counter-factual document will require generating documents which imitate the language production of the advantaged group. E.g. producing a counter-factual for a document written in African-American Vernacular English will require a translation of that document into the other dialect(s) which are present in the dataset.

We also consider fairness through unawareness (Grgic-Hlaca et al., 2016). For NLP tasks, this would mean removing markers that are highly correlated with the classes, which in the case of fake news detection might remove several markers that are indicitive of political leaning rather than whether the statement is fake. We provide this as a baseline.

Another source of discussion was the choice of fairness criteria, we consider demographic parity and equalized odds. Unlike demographic parity, equalized odds does allow for some correlation between the protected attribute and the labels - what it instead requires is similar rates of classification error. However, while demographic parity requires statistical independence, it does not put forth any requirements that the predicted labels are correct. Equalised odds on the other hand requires that the error rates for are constant across protected attributes. We prefer demographic parity for the task of fair fake news detection as we our primary concern is to reduce bias in the weights learned, rather than redistributing predictions based on a fairness criteria.

We apply the cost-sensitive approach proposed by Agarwal et al. (2018) as this does not require a counter-factual scenario, further, unlike many other approaches, the method directly seeks to reweight the classifier such that its bias is mitigated rather than operate on the predictions of the classifier.

5 Conclusion

In this work, we apply the method for re-biasing models according to protected attributes proposed by Agarwal et al. (2018) to the domain of fake news detection on the dataset proposed by **?** using natural language processing. We identify that the dataset proposed by **?** encodes a bias which is learned by models that do not undergo treatment

	"True" label	"False" label		
Unfair	Logistic Regression		Function Words, Cognitive Processes, Affect Words, Time Orientation, Relativity	Social Words, Relativity, Core Drives Needs, Grammar Other, Affect Words
	SVM	Affect Words, Personal Concerns, Time Orientation, Core Drives Needs, Grammar Other	Social Words, Core Drives Needs, Relativity, Time Orientation, Grammar Other	fair Logistic Regression
	Personal Concerns, Relativity, Core Drives Needs, Grammar Other, Function Words	Function Words, Grammar Other, Time Orientation, Relativity, Cognitive Processes		
	SVM	Relativity, Grammar Other, Cognitive Processes, Time Orientation, Function Words	Perceptual Processes, Grammar Other, Time Orientation, Affect Words, Core Drives Needs	

Table 4: LIWC categories in order of frequency as activated by the top 1000 features the Logistic Regression and SVM classifiers.

for potential biases and that this bias in the models can be addressed in two ways: By applying a form of fairness through unawareness (Grgic-Hlaca et al., 2016) in which features which are highly predictive of the protected class are removed from consideration. The other way we show that we can reduce bias is by applying the model proposed by Agarwal et al. (2018) in which we iteratively compute lagrangian multipliers as a function of the violation of our notion of fairness in our model. By applying the method Agarwal et al. (2018), we find that we can reduce biases in our model to the point where it is rendered useless in trying to predict the protected attribute. In our analysis of our model and show that we achieve a 57% reduction in the difference in error between the protected variables. Further, we find that the feature weights examined are modified in small amounts and that by addressing key features significant reductions in bias in the model can be obtained.

In future work, we plan to further investigate the usefulness of this method to reduce bias and seek to identify a method to identify the degree to which a feature should be reweighted to reduce the bias in a classifiers' performance.

Acknowledgments

References

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69, Stockholmsmssan, Stockholm Sweden. PMLR.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4349–4357. Curran Associates, Inc.

Nina Grgic-Hlaca, Muhammad Bilal Zafar, Krishna P Gummadi, and Adrian Weller. 2016. The case for process fairness in learning: Feature selection for fair decision making. In *NIPS Symposium onMachine Learning and the Law*.

Seda Gurses, Rebekah Overdorf, and Ero Balsa. 2018. Pots: The revolution will not be optimized? *CoRR*, abs/1806.02711.

Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. 2018. Multi-source multi-class fake news detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1546–1557, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4066–4076. Curran Associates, Inc.

Yunfei Long, Qin Lu, Rong Xiang, Minglei Li, and Chu-Ren Huang. 2017. Fake news detection through multi-perspective speaker profiles. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 252–256, Taipei, Taiwan. Asian Federation of Natural Language Processing.

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, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426. Association for Computational Linguistics.